UNIVERSIDAD NACIONAL DE LA PLATA

FACULTAD DE INFORMÁTICA

TESINA DE LICENCIATURA

Título: Uso de smartphones para auditar la seguridad de redes inalámbricas
Autores: Bernal, Juan Ignacio – Zurita, Alejandro Enrique
Director: Lic. Venosa, Paula
Codirector: Lic. Lanfranco, Einar
Asesor profesional:
Carrera: Licenciatura en Sistemas – Plan 2007

Resumen

El presente trabajo de grado plantea el desarrollo de una herramienta que permite hacer uso de un smartphone para llevar a cabo la etapa de relevamiento en el marco de un test de penetración de redes inalámbricas dado que las herramientas disponibles no son de gran aceptación por las limitaciones y las complicaciones que presentan para su uso.
Consecuentemente, para el desarrollo de dicha herramienta se procede a realizar un estudio de las soluciones actuales, identificando sus problemáticas, con el objetivo de arribar a la descripción e implementación de una aplicación nativa para el sistema operativo Android, la cual facilita la tarea de relevamiento al auditor.

Palabras Claves

Smartphone, Seguridad, Redes Inalámbricas, Aplicación, Android, Pentest, Root, Cambio de ROM, Webservice, Software Libre

Trabajos Realizados

Se desarrolló una aplicación nativa para Android que permite efectuar la correspondiente etapa de relevamiento de un Pentest de redes inalámbricas.
Llegar al objetivo principal requirió abordar un análisis previo de herramientas preexistentes para poder identificar y entender los problemas que poseen. A partir de este análisis, se presentaron soluciones a las problemáticas identificadas, las cuales fueron incluidas como parte del desarrollo de la aplicación.

Conclusiones

La aplicación desarrollada alcanza los objetivos planteados al inicio de la tesina, representando una solución superadora respecto de las aplicaciones preexistentes. Facilita y hace más seguro el proceso de instalación, hace un uso más responsable de la batería, mejora la usabilidad al incorporar una interfaz nativa de Android y aumenta la portabilidad al utilizar placas inalámbricas de dimensiones reducidas.
En conjunto todas estas mejoras contribuyen a que la solución brinde una mejor experiencia de usuario.

Trabajos Futuros

- Incorporar la posición geográfica de los AP.
- Incorporar nuevos drivers de placas inalámbricas externas para abarcar una mayor cantidad de dispositivos soportados por la aplicación.
- Desarrollar web services que brinden funcionalidad para llevar a cabo el resto de las etapas de un pentest de redes inalámbricas.
- Incorporar otros formatos de presentación de la información obtenida.

Fecha de la presentación: Marzo 2017

Universidad Nacional de La Plata

Facultad de Informática

Uso de smartphones para auditar la seguridad de redes inalámbricas

Tesina de Licenciatura en Sistemas

Autores: Bernal, Juan Ignacio – Zurita, Alejandro Enrique

Director: Lic. Venosa, Paula

Co-Director: Lic. Lanfranco, Einar

La Plata Marzo 2017

A todos aquellos que nos apoyaron y

acompañaron a lo largo de este camino recorrido

Agradecimientos

A la Universidad Nacional de La Plata, pública y gratuita, y en especial a la Facultad de Informática, por permitir desarrollarnos profesional y personalmente durante estos años.

A nuestros directores de tesina, Lic. Paula Venosa y Lic. Einar Lanfranco, por brindarnos su constante confianza y predisposición.

A nuestros padres porque nos impulsaron y apoyaron para realizar la carrera de grado.

A Eugenia Zurita y Juan Pablo Vargas, por ayudarnos en la revisión del informe final de la tesina.

A nuestros amigos y compañeros por el apoyo brindado a lo largo de la carrera de grado.

Juan: A mi compañera.

Alejandro: A mi pareja, compañera de vida.

LICENCIA

Índice

Capítulo 1

1. Introducción

El presente trabajo de grado plantea el desarrollo de una herramienta que permite hacer uso de un smartphone para llevar a cabo la etapa de relevamiento en el marco de un test de penetración de redes inalámbricas dado que las herramientas disponibles no son de gran aceptación por las limitaciones y las complicaciones que presentan para su uso.

Consecuentemente, para el desarrollo de dicha herramienta se procede a realizar un estudio de las soluciones actuales, identificando sus problemáticas, con el objetivo de arribar a la descripción e implementación de una aplicación nativa para el sistema operativo Android, la cual facilita la tarea de relevamiento al auditor.

1.1 Estructura de la Tesina

La Tesina se organiza de la siguiente manera:

- Capítulo 1: Introduce el contexto sobre el cual se desarrolló esta tesina y se plantean los objetivos propuestos.
- Capítulo 2: Presenta el estado del arte y el marco teórico de este trabajo abordando diferentes definiciones que se relacionan con el tema, que ayudan a entender mejor la problemática. También se presentan algunas aplicaciones preexistentes, las cuales se evaluan para poder identificar las problemáticas a tratar.
- Capítulo 3: Define la arquitectura y las tecnologías utilizadas para desarrollar este proyecto.
- Capítulo 4: Describe Pentest Security App, una herramienta desarrollada para realizar tareas de auditoría desde dispositivos móviles.

- Capítulo 5: Presenta las conclusiones en base al objetivo planteado y al trabajo realizado y se detallan trabajos futuros que se desprenden de la presente tesina.

1.2 Motivación

Las redes inalámbricas se han convertido en omnipresentes en el mundo de hoy. Millones de personas las usan diariamente en todos los hogares, oficinas y puntos de acceso públicos para conectarse a Internet y hacer tanto tareas personales como profesionales. A pesar de que las redes inalámbricas, popularmente conocidas como WIFI (1), hacen la vida increíblemente más fácil y nos dan movilidad, estas poseen riesgos.

En los últimos tiempos, las redes inalámbricas inseguras han sido explotadas para entrar en empresas, bancos y organizaciones gubernamentales. La frecuencia de estos ataques se ha intensificado con la creciente popularización de este tipo de tecnología, requiriendo a los expertos desarrollar prácticas y herramientas que permitan su correcto uso (2).

Las auditorías de seguridad de redes inalámbricas resultan significativas al momento de garantizar la integridad y el correcto funcionamiento de los sistemas informáticos. Llevarlas a cabo requiere descubrir los problemas de seguridad mediante pruebas de penetración o Pentest. Las mismas constituyen un procedimiento metodológico y sistemático en el que se simula un ataque real a una red o sistema. El resultado exitoso de este procedimiento depende en gran medida de contar con las herramientas adecuadas y las metodologías sugeridas en los estándares y buenas prácticas (OWASP (3), CEH (4), OSSTMM (5), ISSAF (6), PTES (7)). Así mismo, su realización en tiempo y forma. Demoras en la detección de vulnerabilidades suponen, para los dueños y usuarios de la infraestructura, un aumento en las probabilidades de sufrir un ataque real.

De las tareas que involucra un Pentest de redes inalámbricas, la mayoría pueden realizarse desde la comodidad que brinda una PC o Notebook instalada en un lugar fijo; sin embargo, existen actividades de relevamiento que demandan, a la persona encargada de la auditoría, recorrer espacios físicos descubriendo los Puntos de Acceso (por sus siglas en inglés AP, Access Point

(8)) a los cuales posteriormente se les realizará el diagnóstico. Un ejemplo de esta situación podemos encontrarlo en los edificios pertenecientes a las distintas unidades académicas de la UNLP (9). En estos, la etapa de relevamiento y diagnóstico implica transitar los distintos espacios edilicios, siendo así el uso de una notebook poco práctico y generador de demoras asociadas a la dificultad de su traslado.

Resulta ideal para casos como el detallado anteriormente la utilización de smartphones (término del inglés utilizado para referenciar a los teléfonos inteligentes y dispositivos móviles) que brindan portabilidad. No obstante, las soluciones actualmente disponibles presentan limitaciones y problemáticas ocasionando que no sean de gran aceptación entre los auditores.

Es fundamental entonces, contar con herramientas capaces de adaptarse mejor a los smartphones y al tipo de trabajo, con el objetivo de realizar un diagnóstico más práctico y libre de demoras innecesarias, sin dejar de lado los aspectos de seguridad que deben respetarse en esta tarea.

1.3 Objetivos

El objetivo principal de este trabajo es desarrollar una aplicación nativa para Android que permite efectuar la correspondiente etapa de relevamiento de un Pentest de redes inalámbricas.

Llegar al objetivo principal requiere abordar un análisis previo de herramientas preexistentes para poder identificar y entender los problemas que poseen. A partir de este análisis, se presentan soluciones a las problemáticas identificadas, las cuales son incluidas como parte del desarrollo de la aplicación.

También se implementa un servicio web, que sirve como complemento, para demostrar que las evidencias obtenidas, por la aplicación móvil implementada, articulan correctamente con las tareas relacionadas a la siguiente etapa de un Pentest.

Se espera lograr que las tareas de Pentest de redes inalámbricas, comprendidas por la aplicación desarrollada, representen una alternativa más práctica y cómoda para el usuario que las aplicaciones móviles ya existentes.

Capítulo 2

2. Estado del Arte

En este capítulo se abordarán definiciones y aplicaciones preexistentes relacionadas con el tema, que ayudarán a entender mejor la problemática a tratar. Se empezará por explicar que es un Pentest, para luego hacer foco en su etapa de relevamiento, la cual permitirá establecer el marco teórico necesario para abordar el análisis de las aplicaciones preexistentes y detectar problemáticas asociadas.

2.1 ¿Qué es un Pentest?

Los Test de Intrusión o Pentest (abreviación del inglés, Penetration Test) evalúan los niveles de seguridad de un sistema informático o red mediante la simulación en un entorno controlado, de un ataque por parte de un usuario malicioso conocido comúnmente como hacker. Se aplica un proceso de análisis activo del sistema en busca de posibles vulnerabilidades que podrían resultar de una mala o inadecuada configuración de un sistema, defectos en software conocidos o no, o un fallo de seguridad en un sistema operativo o hardware.

El análisis se realiza desde la posición de un atacante potencial, y puede implicar la explotación activa de vulnerabilidades de seguridad. Los problemas de seguridad que se encuentran se presentan al propietario del sistema junto con una evaluación del impacto que supondría la explotación de las vulnerabilidades halladas dentro de la organización, además de una propuesta de mitigación o una solución técnica.

El propósito de un Pentest es determinar la viabilidad de un ataque y la cantidad de impacto en el negocio de la explotación exitosa, si se descubre. La mejor manera de demostrar la fuerza de una defensa es tratando de penetrar en ella.

Dado que las pruebas de penetración están diseñadas para simular un ataque y utilizar herramientas y técnicas que pueden ser restringidas por la ley, las regulaciones federales, y las políticas de la organización, resulta imprescindible obtener el permiso formal y consensuado del cliente para llevar a cabo las pruebas de penetración.

Este permiso debe estar previamente acordado, organizado y plasmado en un documento físico firmado por ambas partes, donde se indiquen cuáles serán las pautas a seguir, y a partir de aquí empiezan a entrar en juego las fases de un test de intrusión, que se detallan a continuación.

2.2 Fases de un Pentest

A la hora de realizar un Pentest se abarcan un conjunto de etapas, que si no son realizadas en el orden correcto podrían dar lugar a problemas con el cliente que pide la auditoría. Ejemplos de esto sería poner en riesgo el entorno de producción o simplemente vulnerar sin permiso los derechos de la propiedad intelectual y privada de la organización que se está poniendo a prueba (10).

Hay que tener en cuenta que las etapas de un Pentest se pueden ajustar según el tipo de análisis que se desee realizar y las limitaciones impuestas por la organización a ser auditada.

Existen varias metodologías (OWASP, CEH, OSSTMM, ISSAF, PTES) que estandarizan las etapas y el conjunto de tareas a ser realizadas. A partir de ellas, se puede generalizar el proceso de un Pentest en las siguientes fases:

- Reglas del juego: alcance y términos del test de intrusión
- Relevamiento: recolección de la información
- Análisis de vulnerabilidades
- Explotación de las vulnerabilidades
- Post-Explotación del sistema
- Generación de informes

2.2.1 Reglas del juego: alcance y términos del test de intrusión

Al inicio de toda auditoría se debe llegar a un acuerdo sobre los objetivos que el cliente desea alcanzar mediante el test, los límites a los que se verá expuesto el equipo de auditores, es decir, el ámbito de acción, y al carácter confidencial de la información que estará a su disposición. Todo lo acordado deberá ser recogido en un documento firmado por ambas partes donde se declare la conformidad de los responsables del proyecto. Es importante que sea de esta manera, para evitar conflictos futuros respecto a las tareas involucradas en el Pentest.

Normalmente en esta etapa el cliente empieza a entender los peligros a los que se ve expuesta su organización y se concientiza sobre la necesidad de realizar una auditoría de seguridad para preservar sus intereses.

2.2.2 Relevamiento: recolección de la información

Esta es la primera etapa práctica del Pentest, donde el equipo de auditores hará uso de diferentes técnicas para obtener la mayor cantidad de información sobre la organización que se somete al test.

Se realizan investigaciones tratando de recolectar información pública sobre la plataforma tecnológica del cliente, utilizando para ello técnicas pasivas y activas de relevamiento de información.

Este procedimiento permite empezar a delimitar las áreas sobre las que luego se focalizará la siguiente etapa, es decir se define el alcance de la auditoría.

2.2.3 Análisis de vulnerabilidades

Después de recolectar toda la información disponible mediante las técnicas adecuadas, se prosigue a analizar y organizar todos los resultados con la finalidad de encontrar vulnerabilidades. Como resultado se obtiene un modelo con toda la información extraída y las inconvenientes detectados, a partir del cual se decidirá cuáles son las vulnerabilidades que se desean explotar teniendo en cuenta las prioridades del cliente, el riesgo asociado al negocio que poseen las mismas y el tiempo para realizar las tareas de explotación.

Una vez seleccionadas las falencias sobre las que se trabajará, se elabora un plan de acción donde se planifica el orden y método de explotación para cada una de las vulnerabilidades.

2.2.4 Explotación de las vulnerabilidades

Habiendo definido un plan de acción en la etapa anterior, los auditores comienzan la fase de explotación de las problemáticas seleccionadas. Para llevarla a cabo, deben tener en cuenta la información recopilada anteriormente y su experiencia profesional para el manejo de exploits (11) y herramientas que permitirán lograr los objetivos perseguidos por el plan de acción.

La falta de experiencia del auditor o errores en la recolección de información de las etapas anteriores pueden ocasionar falsos positivos, que aumentan el tiempo empleado para la auditoría y ponen en duda la veracidad o fiabilidad del proceso que se ha empleado para realizar dicha recolección de información.

Todo ataque posee consecuencias, las cuales deben ser anticipadas y dimensionadas por los auditores, para evitar aquellas que podrían perjudicar a la organización que se está auditando. Es importante entonces asegurar que, en todo momento, se posea el control de las acciones que se están llevando a cabo, lo cual muchas veces implica no utilizar herramientas automatizadas y abordar las tareas en forma manual, lo que permitirá garantizar la veracidad de los resultados positivos encontrados y evitar consecuencias negativas causadas por el exceso o mal uso de los exploits de las herramientas autogestionadas. Un ejemplo de esta última sería el bloqueo o caída de un proceso crítico, causado por el abuso de las técnicas de explotación de vulnerabilidades del sistema.

2.2.5 Post-Explotación del sistema

Una vez lograda la etapa anterior, y en el caso de haber podido explotar con éxito algunas de las vulnerabilidades encontradas, se puede realizar un nuevo proceso de búsqueda con el fin de obtener mayor cantidad de información.

Definir esta segunda ronda de explotación estará condicionada por el acuerdo de auditoría mencionado en la primera etapa, como así también por el tiempo, los recursos y los riesgos que conlleva para la organización llevar a cabo esas tareas.

2.2.6 Generación de informes

Esta es la última fase, y es considerada la parte más importante del test, donde se informa al cliente sobre las pruebas que se han realizado y los resultados obtenidos de cada una de ellas.

Entre la información que forma parte del documento final de la auditoría debe incluirse, para cada una de las tareas que se han realizado, las técnicas, las herramientas utilizadas, los procedimientos operativos, las vulnerabilidades que se han descubierto y el nivel de gravedad, conocido como nivel de riesgo, que supone para la seguridad de la organización. Así mismo, teniendo en cuenta que los destinatarios de dicho informe pueden tener diferentes perfiles, es aconsejable desglosar el mismo en dos entregables: ejecutivo y técnico.

En el informe técnico se documentará toda la información con un alto nivel de detalle y recomendaciones para sus posibles soluciones, con la finalidad de que a partir de su interpretación, el equipo técnico responsable dentro de la organización pueda implementar las soluciones propuestas u otras alternativas a fin de contrarrestar las falencias encontradas.

En el informe ejecutivo se documentarán las vulnerabilidades halladas, con un mínimo detalle técnico, haciendo énfasis en los riesgos que suponen para la organización. Al igual que en el informe técnico, se describirán las soluciones y se propondrá un plan de acción para poder priorizar la solución de las mismas.

2.3 ¿Por qué es necesario realizar un Pentest de redes inalámbricas?

Las redes inalámbricas son una extensión del perímetro de la infraestructura de las organizaciones. Fallas en la seguridad de las mismas, pueden representar un riesgo. Un ejemplo son los puntos de acceso de tipo Rogue (12)(AP no autorizados), instalados por los empleados, que al no seguir las directrices de redes de la organización, pueden comprometer la seguridad de la misma.

Existen muchos casos donde las organizaciones sufren incidentes que podrían haberse evitado, si los mecanismos de protección hubieran sido reforzados en su momento. Los incidentes comprenden sucesos tales como fuga

de información, accesos no autorizados, pérdida de datos, entre muchos otros. El análisis de los mecanismos de protección debe ser una tarea proactiva permitiendo al pentester (10)(persona que lleva adelante la auditoría) encontrar las vulnerabilidades dentro de los mismos y brindar una solución antes de que un ciberdelincuente aproveche esta debilidad (13).

Con tal motivo, al realizar un Pentest se persiguen los siguientes objetivos (14):

- Probar que las medidas de seguridad hayan sido diseñadas e implementadas de forma correcta.
- Identificar vulnerabilidades.
- Identificar el riesgo real de una vulnerabilidad. Por ejemplo, varias vulnerabilidades de riesgo bajo pueden alinearse para, en su conjunto, construir una vulnerabilidad de alto riesgo.
- Demostrar que el peligro es real. Muchas veces la seguridad es subestimada en la toma de una decisión estratégica y operacional de una compañía. La única forma de lograr un cambio de actitud es mediante la demostración de un ataque inocuo (mismo efecto que una vacuna).

2.4 Pentest de redes inalámbricas y la etapa de relevamiento

Como se mencionó al comienzo de esta tesina, el tema central de estudio está relacionado con la etapa de relevamiento de un Pentest de redes inalámbricas, cuyo objetivo es poder recolectar la mayor cantidad de información posible de la infraestructura de red a ser auditada.

Conocer cuáles son las tareas que se realizan en esta etapa, será de ayuda para proyectar los criterios de selección a tener en cuenta en el análisis de la sección siguiente.

A continuación se describen algunas de las tareas más comunes realizadas en esta etapa (15):

- Descubrir AP de tipo rouge: Consiste en verificar si existen APs que estén conectados sin autorización a la red de la organización.
- Descubrir AP ocultos: Consiste en verificar si existen APs configurados con SIDD oculta.
- Detectar cantidad de dispositivos asociados a un AP: Se analizan los paquetes capturados de la red inalámbrica para determinar el balance de carga de cada AP.
- Capturar el tráfico entre el AP y sus dispositivos asociados: Se capturan los paquetes de la red para estudiar el tipo de tráfico que están transmitiendo.
- Verificar la encriptación: Se verifica el tipo de encriptación configurado en cada AP.
- Detectar generación rápida o excesiva de tráfico: Se analiza el tráfico de la red para detectar posibles ataques de denegación de servicio.
- Obtener el Handshake WPA (16)/WPA2 (17): Consiste en obtener los paquetes relacionados con el handshake.

2.5 Tecnologías existentes

En esta sección se llevará a cabo un análisis de las tecnologías existentes con la finalidad de poder detectar problemáticas asociadas a las mismas. Se comienza por establecer un conjunto de criterios de selección, evaluación y presentación para las aplicaciones.

Como criterios de selección, se decidió contemplar las 5 aplicaciones más renombradas, preferentemente de uso y código libre, creadas desde el lanzamiento del Iphone (18) hasta la actualidad, que tengan capacidad de habilitar el modo monitor (19) de una placa inalámbrica y permitan hacer algunas de las tareas correspondientes a la etapa de relevamiento.

Para presentarlas se decidió ordenarlas cronológicamente por fecha de lanzamiento, mencionando una breve descripción, una reseña histórica, un análisis técnico funcional y las desventajas encontradas.

Como criterios de evaluación, la reseña histórica tendrá como objetivo dar a conocer el origen y contexto de la aplicación, mientras que el análisis técnico funcional pondrá énfasis en descubrir las dificultades que posee la aplicación para obtener el modo monitor y hacer uso de sus funcionalidades. A tal fin se abordará, de cada aplicación, el proceso de instalación, la forma de uso, los resultados de la ejecución y sus funcionalidades. Es importante resaltar que el objetivo no será describir una guía detallada de pasos a seguir para su instalación o uso, sino describir los aspectos más importantes de la experiencia de uso obtenida. Una vez concluido el análisis, se mencionan las problemáticas generales encontradas.

Como resultado de aplicar los criterios de selección, se tomaron como caso de estudio las siguientes aplicaciones:

- Kismet para Nokia N900
- BcMon
- Android PCAP (kismet)
- PwnAir Pro (interface gráfica para el Aircrack-ng for Android)
- Android Open Pwn Project (AOPP) (PWM-Phone)

2.5.1 Kismet para Nokia N900

2.5.1.1 Descripción

Esta solución open source está comprendida por un driver modificado, que habilita el modo monitor del Nokia N900 (20), y una aplicación llamada Kismet (21) que fue portada para el dispositivo con el objetivo de poder realizar tareas de auditoría de redes inalámbricas.

2.5.1.2 Reseña Histórica

Tras el boom originado por el lanzamiento del Iphone (18) en Junio del 2007, empezó la lucha entre empresas por ganar el creciente mercado de los smartphones. Nokia (22), compañía que lideraba la telefonía móvil en ese entonces, apostó a esta nueva tendencia lanzando un teléfono que pudiera competir con el Iphone. Así fue como en noviembre del 2009 se anunció el Nokia N900 (23).

El Nokia N900 se incorporó con novedades que prometían un teléfono potente para la época. Contaba con un procesador ARM de 600 mhz, 256MB de memoria RAM y un sistema operativo open source, basado en Linux, llamado Maemo (24). Estos factores, atrajeron a entusiastas de Linux que adquirieron el teléfono a fin de extender su funcionalidad. Así fue el caso del Austriaco David Gnedt, quien en Abril del 2010 (25)(solo 4 meses después de la fecha de lanzamiento del N900) generó un precedente en la telefonía móvil, al portar Kismet para que funcione en Maemo y desarrollar un parche que permitía poner en modo monitor la placa Wifi, con chip wl1251, que posee el Nokia N900. Estos dos aportes marcaron el primer caso documentado que permite realizar tareas de auditoría en redes inalámbricas a través de un smartphone.

Sin embargo, en forma paralela a Maemo, se estaba desarrollando el sistema operativo Android, el cual tomó más importancia en el mercado de los dispositivos móviles a partir de que la empresa Samsung pusiera a la venta, en junio del 2010, el smartphone "Galaxy S1" (26).

Samsung ocasionó un gran impacto en el mercado mobile, vendiendo 5 millones de unidades (26) en tan solo 4 meses, mientras que Nokia no pudo superar las 100 mil ventas de su modelo N900 a pesar de salir al mercado casi 7 meses antes (27).

Finalmente Nokia discontinuó el desarrollo de Maemo tras liberar su última versión en octubre del 2011 (28).

La solución propuesta por David Gnedt, aplicable solamente al N900 y con limitaciones en su uso, se vio afectada por lo anteriormente mencionado, dando como resultado que no se popularice y quede descontinuada.

2.5.1.3 Análisis técnico funcional

A continuación se detallan aspectos relacionados con la aplicación. Si bien no se pudieron realizar pruebas propias, debido a que no fue posible conseguir un ejemplar del teléfono mencionado, la información proporcionada por el blog de su creador (29), videos en internet (30) (31) y la experiencia propia de haber interactuado con otros dispositivos móviles permitió explicar las secciones que se detallan a continuación.

2.5.1.4 Procedimiento de instalación

Para hacer uso de Kismet y del modo monitor, hay que habilitar un modo especial en el gestor de paquetes de Maemo, que permite descargar paquetes de terceros y modificar el sistema operativo. Esta funcionalidad, nombrada como "extras-devel repository", requiere un usuario experimentado y mucho cuidado al momento de generar cambios, dado que se puede provocar alteraciones que dejen fuera de uso al dispositivo móvil.

Esta funcionalidad se utiliza para generar los siguientes cambios del sistema operativo:

- Descargar e instalar el "Enhanced Linux kernel for power users" que contiene el kernel modificado con el driver en modo monitor. Una vez hecho esto, la placa inalámbrica queda habilitada para entrar en dicho modo, hasta que se vuelve a cargar el kernel normal.
- Descargar e instalar "Kismet".

2.5.1.5 Forma de uso

Para poder usar Kismet hay que seguir los siguientes pasos:

- Abrir un terminal/consola.
- Loguearse como usuario root para obtener todos los privilegios.
- Iniciar la aplicación escribiendo en la consola "kismet".

2.5.1.6 Resultados de la ejecución y funcionalidad

La aplicación Kismet presenta una interfaz de consola muy limitada en aspectos visuales, brindado poca información al usuario. La misma se puede apreciar en la figura 2.1.

Con Kismet y la placa en modo monitor se puede generar un pcap del tráfico entre los diferentes APs y Stations detectados, para poder ser analizado con posterioridad.

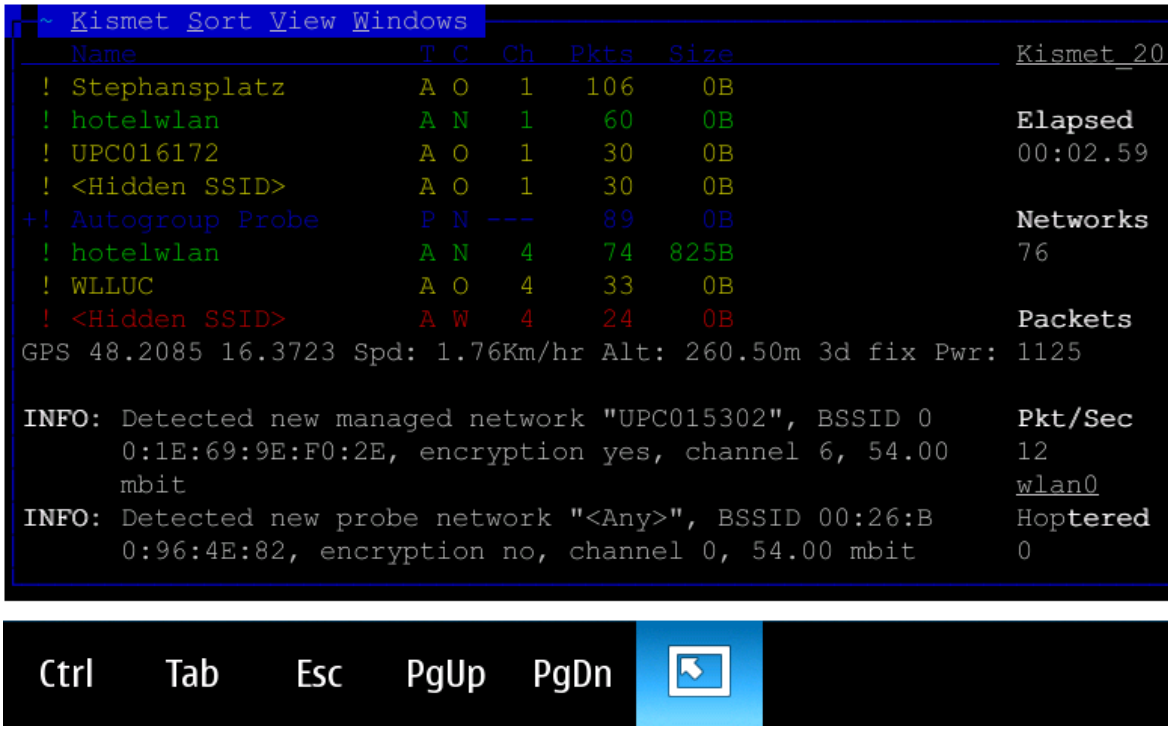


Fig. 2.1 – Kismet para Nokia N900

2.5.1.7 Desventajas

Teniendo en cuenta lo anteriormente comentado se encontraron las siguientes desventajas:

- Habilitar "extras-devel repository" y el modo Root pone en riesgo la estabilidad y seguridad del sistema operativo.
- La aplicación no permite hacer un ataque de desautenticación (10) para poder obtener los handshake WPA2.
- Requiere un usuario avanzado para poder instalar y usar la aplicación.
- El modo monitor inhabilita el gestor de energía del teléfono, ocasionando que la batería dure muy poco.
- El modo monitor inhabilita el uso de la placa wifi para el resto de las aplicaciones y para conectarse a un AP en forma normal.
- El modo monitor genera mal funcionamiento de la conexión Bluetooth.

2.5.2 BcMon

2.5.2.1 Descripción

Bcmon es una aplicación open source para Android que integra un conjunto de herramientas de la suite de Aircrack-ng y un parche para habilitar el modo monitor de las placas wifi marca Broadcom. En conjunto permiten realizar tareas de auditoría de redes inalámbricas.

2.5.2.2 Reseña histórica

Como se mencionó anteriormente, en forma paralela a Maemo, se estaba desarrollando Android, sistema operativo basado en Linux, el cual tomó más importancia, en el mercado de los dispositivos móviles, a partir de que la empresa Samsung pusiera a la venta, en junio del 2010, el smartphone "Galaxy S1" el cual incorporaba Android en su versión 2.1 (32).

Android siguió creciendo y ganando popularidad, lo cual cautivó a muchos desarrolladores que empezaron a crear soluciones compatibles con el sistema. Así fue el caso del grupo conformado por Omri Ildis, Ruby Feinstein y Yuval Ofir, que en septiembre del 2012 implementaron y liberaron la aplicación BcMon y un parche que pone en modo monitor las placas inalámbricas con chipset Broadcom (33). Estos dos aportes marcaron el primer caso documentado para Android, que permite activar el modo monitor para realizar tareas de auditoría.

Inicialmente el parche se desarrolló para los teléfonos Google Nexus One y Samsung Galaxy S2 que poseen placas Inalámbricas internas marca Broadcom con chipset bcm4329 y bcm4330 respectivamente. Posteriormente la comunidad de Android utilizó el parche en dispositivos que poseen chipsets broadcom compatibles.

Para poder aplicar el parche, que modifica el módulo del kernel que contiene el driver de la placa inalámbrica, es necesario tener acceso Root y tener instalado la ROM CyanogenMod (34).

La solución propuesta tuvo gran difusión, aunque se ve limitada a dispositivos que tengan chipset Broadcom 4329 o 4330 (Samsung Galaxy S1, Samsung Galaxy S2, Nexus 7, Huawei Honor). No es replicable en dispositivos actuales debido a que utilizan otros chipset, de los cuales no hay documentación para desarrollar su driver con modo monitor.

Los smartphones compatibles con la solución, empezaron a quedar en desuso debido a su antigüedad, siendo reemplazados por otros de mejor rendimiento, prestaciones y nuevas versiones de Android.

Si bien con el pasar de los años la comunidad de Cyanogenmod, fue portando las nuevas versiones de Android y del parche a los dispositivos mencionados, con el fin de actualizarlos para prolongar su vida útil, actualmente se ha discontinuado su desarrollo debido a la baja cantidad de usuarios que los utilizan.

2.5.2.3 Análisis técnico funcional

A continuación se detallaran aspectos relacionados con la instalación y uso de la aplicación, obtenidos a partir de pruebas propias realizadas en un dispositivo Samsung Galaxy S2. Se han tomado como referencia la información proporcionada por el blog de BcMon (35), tutoriales (36) y videos en internet (37).

2.5.2.4 Procedimiento de instalación

Para hacer uso de BcMon y habilitar el modo monitor en los chipset Broadcom, se debe rootear el teléfono e instalar la ROM Cyanogenmod.

Realizar estas modificaciones requiere adquirir previamente conocimientos sobre cómo llevarlas a cabo, dado que existen riesgos implícitos asociados a tales prácticas que pueden dejar el teléfono inutilizable.

Antes de empezar se mencionan algunos consejos a tener en cuenta:

- Es importante empezar por verificar que el dispositivo a modificar cuente con la batería sana y esté totalmente cargada, dado que el proceso de cambio de ROM anula temporalmente el gestor de carga de la batería y un corte en el suministro de energía, mientras se está llevando a cabo la modificación, normalmente deja al teléfono en un estado inconsistente y difícil de recuperar.
- Verificar siempre que la versión de ROM y recovery a instalar sean compatibles con el dispositivo móvil. Dada la gran cantidad de variaciones comerciales que hay del Samsung Galaxy S2, y la generalidad de los tutoriales que se pueden encontrar en Internet, es muy común

terminar con una instalación fallida, causada por la incompatibilidad de los componentes.

- Muchos fabricantes implementan los llamados "efuse" (38), que funcionan como fusible digital y se "rompen" al cambiar el bootloader, impidiendo que se pueda instalar una versión inferior a la actual. Es importante tenerlo en cuenta, dado que muchas ROMs alternativas requieren instalar versiones superiores del mismo, lo cual tiene como consecuencia que no se pueda revertir el procedimiento.
- El proceso de instalación no se puede pausar, tarda varios minutos, consume bastante energía y procesamiento, generando un aumento de la temperatura del dispositivo. Es aconsejable estar pendiente del mismo para evitar posibles interrupciones o fallas de la instalación ocasionadas por el exceso de temperatura o cortes del suministro eléctrico.
- Para disminuir la probabilidad de instalaciones fallidas o problemas de rendimiento, se aconseja, antes de empezar con las modificaciones, hacer un resguardo de la información propia y restaurar el teléfono a sus valores de fábrica. Esto evitará que configuraciones anteriores interfieran con la nueva ROM a instalar.

El proceso de rooteo consiste en poder obtener permisos de superusuario, también llamado usuario root, los cuales habilitan la edición de archivos protegidos del sistema operativo. Hay que tener en cuenta que no existen herramientas oficiales de Samsung para tal fin (39), por lo que obtenerlo requiere de buscar, descargar y probar software de terceros, que mediante diferentes técnicas (explotación de vulnerabilidades) obtienen el acceso root requerido. Ninguno de estos procedimientos tiene garantía respecto a las consecuencias por su instalación, contenido y modificaciones que producen, por lo que merece tomar precauciones respecto a su uso.

Para llevar a cabo el root se probaron tres tipos de alternativas. A continuación se hace una breve mención de ellas y de la experiencia obtenida:

- Aplicación de usuario Kingo Root APK (40) o z4root (41): Este tipo de aplicaciones se instalan y ejecutan como una aplicación normal de Android. Durante las pruebas, los intentos de obtener el acceso root fueron más propensos a fallar cuando la versión de Android no era compatible. Esto se debe a que muchas veces Samsung varía ligeramente

las versiones de su sistema operativo, según la empresa de telefonía móvil que vende el producto.

- Aplicación para recovery (42): son aplicaciones de tipo script y vienen empaquetadas en un formato .zip compatible con el de recovery del teléfono, que exige que estén firmadas digitalmente por la empresa creadora del mismo. Dado que Samsung no proporciona estos paquetes, queda en evidencia que la firma digital, si bien funciona, no es legítima. Para su funcionamiento, requiere copiar el script en la SD del dispositivo e iniciar en modo recovery para poder instalarlo. La tasa de fallos ha sido alta en comparación con los otros métodos, por lo que no se recomienda.
- Script ejecutable con ADB: Este procedimiento permite rootear el teléfono conectándolo a una PC de escritorio y ejecutando un script que hace uso de la aplicación Android Debug Bridge (ADB) (43), para comunicarse con el smartphone. Si bien el proceso suele funcionar, requiere instalar los drivers del dispositivo en la computadora a utilizar, habilitar el modo depurador en el teléfono y ejecutar el script correspondiente. Este método es el que mejor funcionó, aunque la aplicación ADB no siempre detectaba el teléfono.

Una vez obtenido el root se procede a instalar el recovery alternativo, el cual permite instalar y bootear la ROM personalizada. Existen dos tipos de recovery: El clockworkmod (CWM) (44) y el Team Win Recovery Project (TWRP) (45). Son prácticamente iguales, aunque TWRP cuenta con una interfaz gráfica más amigable basada en botones y menús. Estos se pueden seleccionar desde la pantalla táctil, mientras que CWM está basada en menús y opciones seleccionables desde los botones físicos, como los de volumen y el de encendido.

Para llevar a cabo el proceso de instalación se hace uso del programa Odin, el cual permite cargar y sobrescribir la partición de recovery del teléfono por la imagen alternativa deseada. El proceso consiste en apagar el teléfono, prenderlo en modo “FastBoot”, conectarlo a la computadora donde se tiene instalado Odin, cargar los drivers correspondientes y ejecutar el programa mencionado seleccionando el recovery alternativo. Una vez terminado el proceso, el celular se reiniciara automáticamente y si la instalación fue exitosa, el booteo terminará con la carga del sistema operativo Android. Si la instalación no fue exitosa, el

celular entrará en un loop, reiniciándose a los pocos segundos de haber iniciado. En ese caso se deberá volver a cargar el firmware Stock mediante Odin.

Se seleccionó e instaló el recovery TWRP por disponer de una interface más amigable, la cual facilitó las tareas posteriores.

Como ROM alternativa se utilizó CyanogenMod v9 que contiene el driver modificado con modo monitor para la placa inalámbrica. Se empieza por copiar la imagen de la ROM a la tarjeta SD, bootear el modo recovery del TWRP, borrar la partición del sistema operativo /system (aconsejable también el /data y /cache) y seleccionar la instalación del archivo cargado en la SD. El proceso se realizará en forma automática, y al terminar se debe reiniciar el dispositivo, el cual cargará el nuevo sistema operativo.

2.5.2.5 Forma de uso

Para poder usar BcMon hay que ejecutar la aplicación y otorgarle permisos de Root, de lo contrario no inicia. Luego se debe habilitar el driver modificado y el modo monitor para poder hacer uso de las herramientas de consola.

BcMon integra las siguientes aplicaciones de consola:

- **Airodump:** Airodump-ng se usa para capturar paquetes wireless 802.11 (1) y es útil para ir acumulando vectores de inicialización IVs con el fin de intentar usarlos con Aircrack-ng y obtener la clave WEP (46).
- **Wash:** se utiliza para saber si un router tiene habilitado WPS (1), ganando tiempo para el uso de aplicaciones como reaver (47).
- **Besside-ng**: Permite crackear en forma automática redes WEP (1) y WPA.

2.5.2.6 Resultados de la ejecución y funcionalidad

Bcmon presenta una interfaz basada en menús. No obstante las herramientas contenidas por esta aplicación (airodump, wash, besside-ng) poseen su interfaz de consola original, con aspectos visuales limitados, que brindan poca información al usuario.

La misma se puede apreciar en la imagen Fig. 2.2 y Fig. 2.3.

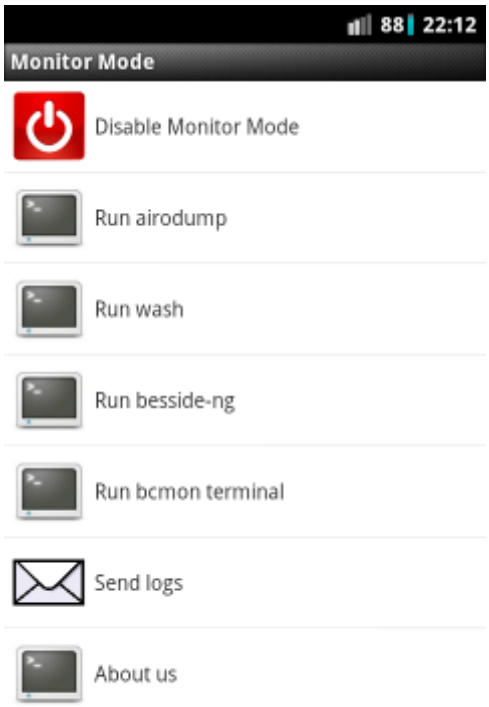


Fig. 2.2 –Bcmon

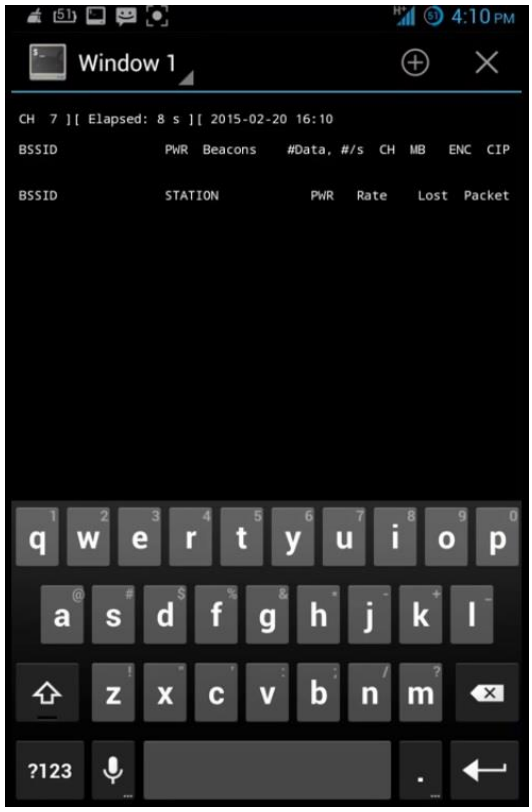


Fig. 2.3 –Bcmon

2.5.2.7 Desventajas

Teniendo en cuenta lo anteriormente comentado, se encontraron las siguientes desventajas:

- Se requiere rootear el teléfono para poder usar la aplicación.
- Se requiere cambiar el firmware original por la ROM alternativa CyanogenMod.
- Requiere un usuario avanzado para poder instalar y usar la aplicación.
- La interfaz de consola no resulta práctica para trabajar.
- El modo monitor inhabilita el uso normal de la placa wifi para el resto de las aplicaciones, impidiendo conectarse a un AP.
- El proceso de rooteo y cambio de ROM puede llegar a durar 4 hs.

2.5.3 Android PCAP

2.5.3.1 Descripción

Android PCAP es una aplicación de código libre que hace uso de la API Android USB Host para poder comunicarse con una placa inalámbrica USB y hacer capturas en formato pcap del tráfico inalámbrico. Luego, estos pueden ser analizados con aplicaciones externas como WireShark (48), Eye P.A (49) y Kismet.

Se caracteriza por no requerir de permisos de superusuario (root) o cambios en el firmware del teléfono.

2.5.3.2 Reseña Histórica

El lanzamiento de la aplicación BcMon fue el punto de partida para poder realizar tareas de auditoría de redes inalámbricas desde Android, pero dado sus requisitos de instalación (rootear el teléfono y cambiar el firmware por uno alternativo), se ve limitada a unos pocos dispositivos y a usuarios avanzados que estén dispuestos a modificar su teléfono para tal fin.

Teniendo en cuenta estas limitaciones, Mike Kershaw, desarrollador del proyecto "kismetwireless.com" propuso un enfoque diferente para poder obtener el modo monitor de una placa de red inalámbrica: utilizar la API USB Host de Android, para comunicarse con una placa inalámbrica USB externa. Con esto en mente, eligió una placa de red y portó su driver de Linux a Java para que pudiera ser ejecutado en el espacio de usuario de una aplicación de Android. De esta manera se evita tener que rootear y alterar el kernel original con un driver de Linux alternativo. La aplicación se llamó Android Pcap y fue liberada en diciembre del 2012. Para poder utilizarla se requiere tener un teléfono con Android 4.0 o superior, una placa inalámbrica USB con chipset RTL8187 y un cable USB OTG. En particular, la aplicación solo soporta los siguientes modelos comerciales de placas inalámbricas (50):

- Alfa AWUS036H
- LevelOne WNC-0301USB v5
- LevelOne WNC-0305USB
- AirLive WL-1600USB
- NETGEAR WG111 v2 y v3

La aplicación es fácil de instalar y utilizar, sin embargo presenta una gran desventaja: la placa inalámbrica externa soportada tiene un elevado consumo eléctrico provocando el rápido agotamiento de la batería interna e incompatibilidad con muchos smartphones que no son capaces de entregar la corriente necesaria para su correcto funcionamiento. A su vez, el gran tamaño físico que poseen las placas inalámbricas con el chipset mencionado, dificultan su uso y portabilidad.

Las desventajas mencionadas anteriormente, sumado a la acotada funcionalidad de la aplicación, que solo permite generar Pcap, ocasionaron que la misma no fuera de gran utilidad y se descontinuara.

2.5.3.3 Análisis técnico y funcional

A continuación se detallaran aspectos relacionados con la instalación y uso de la aplicación mencionada obtenidos a partir de las pruebas propias realizadas en un Motorola G primera generación. Se han tomado como referencia la información proporcionada por la página oficial de la aplicación (51), tutoriales y videos en internet (52) (53) (54).

2.5.3.4 Procedimiento de instalación

Para poder instalar Android Pcap se requiere tener un dispositivo con Android 4.0 o mayor que soporte USB OTG y descargar e instalar la aplicación desde el Play Store.

2.5.3.5 Forma de uso

Actualmente las placas inalámbricas mencionadas son muy antiguas y no se encuentran disponibles comercialmente, motivo por el cual no se pudo realizar la prueba de captura de paquetes. No obstante, la instalación y la inspección de la aplicación, complementadas con pruebas realizadas por terceros que muestran la captura de paquetes, permiten entender el completo funcionamiento de la misma.

2.5.3.6 Resultados de la ejecución y funcionalidad

La aplicación se caracteriza por su sencillez. Permite seleccionar los canales de redes inalámbricas a ser analizadas, capturar el tráfico de las redes detectadas y generar un pcap con los resultados. Su interfaz está diseñada para el uso en dispositivos móviles, sin requerir de pantallas tipo consola. En las figuras 2.4 y 2.5 se puede apreciar su interfaz.

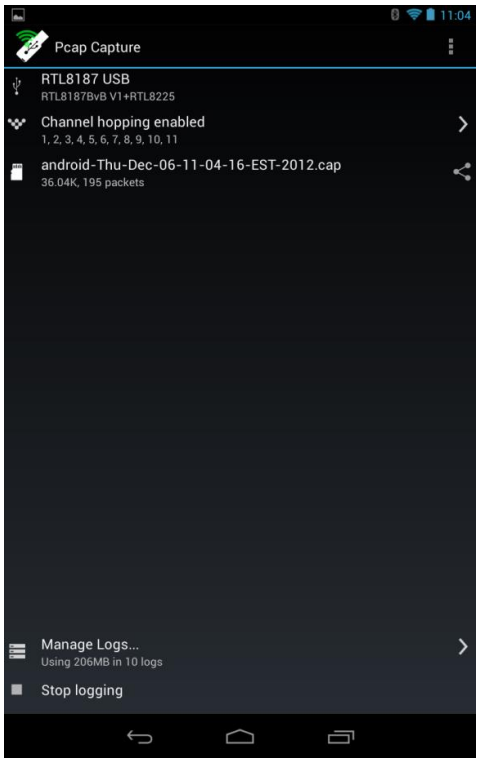


Fig. 2.4 –Android PCAP

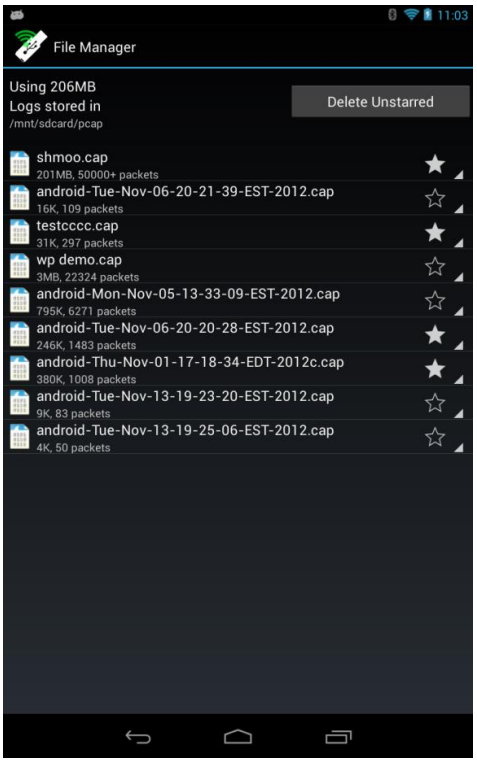


Fig. 2.5 – Android PCAP

2.5.3.7 Desventajas

Teniendo en cuenta lo anteriormente comentado se encontraron las siguientes desventajas:

- Solo es compatible con algunas placas inalámbricas USB que poseen el chipset RTL8187.
- Las placas inalámbricas USB compatibles son muy antiguas y ya no se encuentran comercialmente disponibles.
- En algunos dispositivos, el puerto USB no es capaz de entregar la corriente necesaria para el correcto funcionamiento de las placas inalámbricas compatibles con la aplicación.
- Las placas inalámbricas USB compatibles poseen un alto consumo eléctrico, agotando rápidamente la batería del dispositivo.
- La aplicación no permite hacer un ataque de desautenticación para poder obtener los handshake WPA2.

2.5.4 PwnAir Pro

2.5.4.1 Descripción

PwnAir Pro es una aplicación para Android que integra la aplicación BcMon con una interfaz gráfica más amigable basada en menús. Permite realizar tareas de auditoría de redes inalámbricas.

2.5.4.2 Reseña histórica

El lanzamiento de la aplicación BcMon dio lugar a que otros miembros de la comunidad de Android empezarán a desarrollar aplicaciones que hicieran uso de su funcionalidad. Una de la aplicaciones que refleja este caso fue PwnAir, creada y liberada en mayo del 2014 por KrisWebDev, que brinda una interfaz de usuario mejorada a las herramientas de la suite de Aircrack que integran Bcmon, haciendo más práctico su uso.

Si bien PwnAir estaba basada en Bcmon, solo es capaz de habilitar el modo monitor en el Samsung Galaxy S1. KrisWebDev publicó la aplicación para que pueda ser usada libremente, pero nunca liberó el código fuente para que se puedan hacer mejoras o soportar otros dispositivos. Con el tiempo, la aplicación dejo de ser mantenida y terminó en desuso.

2.5.4.3 Análisis técnico funcional

A continuación se detallarán aspectos relacionados con la instalación y uso de la aplicación mencionada obtenidos a partir de las pruebas propias realizadas en un Samsung Galaxy S2 y un Motorola G primera generación. Se han tomado como referencia la información proporcionada por la página oficial de la aplicación, tutoriales y videos en internet.

2.5.4.4 Procedimiento de instalación

PwnAir al igual que BcMon, requiere rootear y cambiar el sistema operativo original. En el caso de PwnAir, solo es compatible con la ROM, basada en CyanogenMod v9, que posee el kernel “Aries” con el driver modificado para modo monitor. Una vez hecho lo anterior, se instala la APK de PwnAir.

Dado que no se pudo conseguir un Samgung Galaxy S1 (único dispositivo soportado), se intentó probar la aplicación en el Samsung Galaxy S2, modificado anteriormente para instalar Bcmon, pero los resultados no fueron satisfactorios.

Se probó también instalando la aplicación en un Motorola G primera generación sin modificar, obteniéndose el mismo resultado que en el Samsung Galaxy S2: se puede ejecutar la aplicación pero, al tratar de habilitar el modo monitor para trabajar con la placa inalámbrica, la aplicación se detiene presentando un error.

2.5.4.5 Forma de uso

Una vez iniciada la aplicación, se visualiza un menú dividido en 4 secciones. Se empieza por "Load", la cual permite habilitar y deshabilitar el modo monitor de la placa wifi. Una vez habilitado, se procede a ir a la solapa "Target", donde se podrá iniciar la búsqueda de APs. Una vez detectados, se podrá seleccionar uno para empezar a capturar su tráfico. En la siguiente solapa, llamada "Attack" se realiza el ataque para obtener los paquetes necesarios para el proceso de crackeo a realizar en la siguiente sección. En esta última, llamada "Crack" se lleva a cabo el proceso de crackeo para tratar de obtener la contraseña WEP o WPA del AP analizado.

2.5.4.6 Resultados de la ejecución y funcionalidad

La aplicación utiliza componentes visuales (botones y menús) como método de entrada para la carga de opciones y la consola como método de salida para reflejar las tareas que se están llevando a cabo. Esto resulta en una interfaz más amigable que la proporcionada por la consola de Bcmon.

En las figuras 2.6, 2.7, 2.8 y 2.9 se puede apreciar su interfaz.

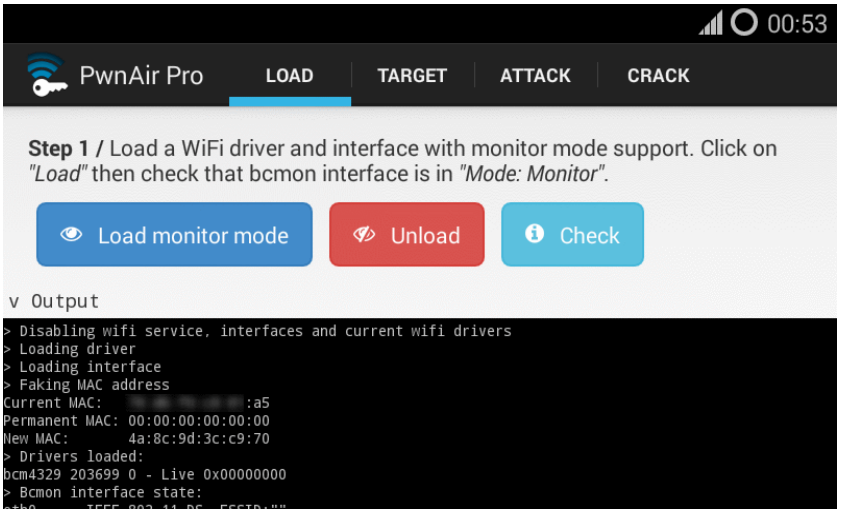


Fig. 2.6 – PwnAir Pro

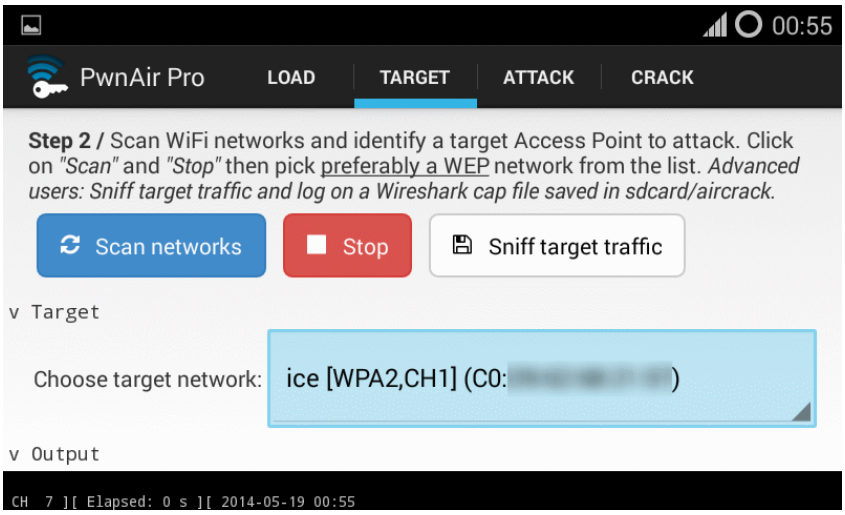


Fig. 2.7 – PwnAir Pro

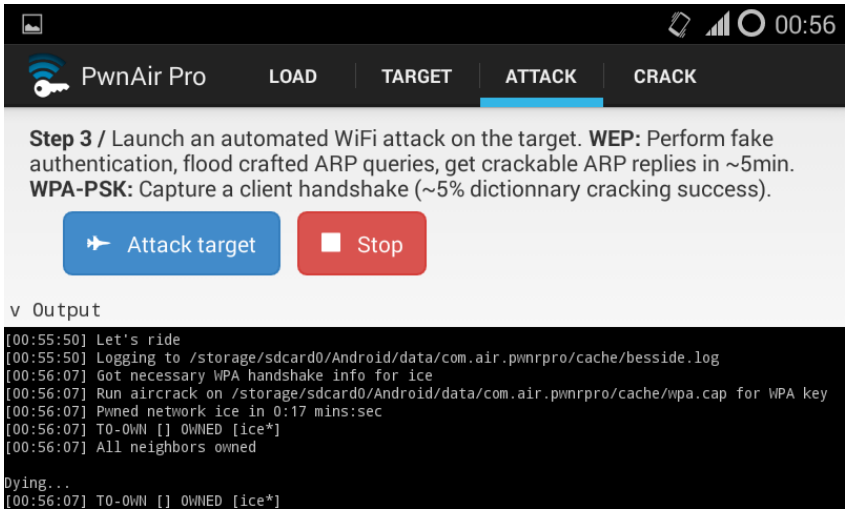


Fig. 2.8 – PwnAir Pro

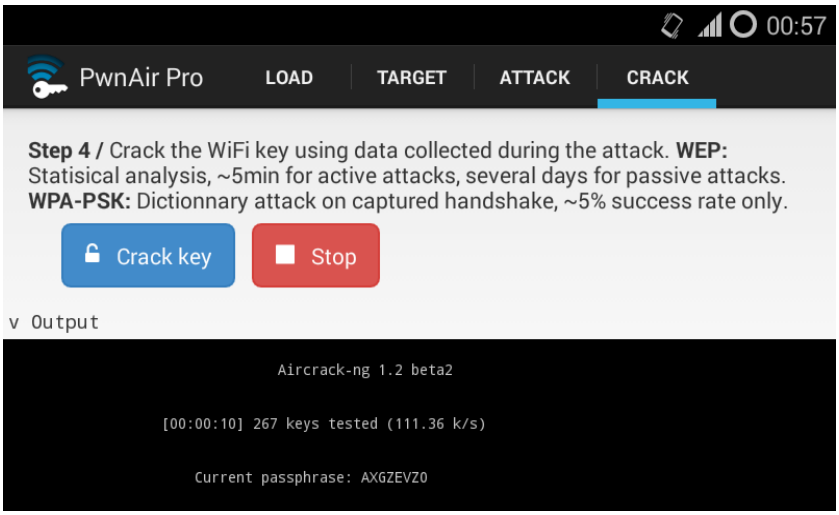


Fig. 2.9 – PwnAir Pro

2.5.4.7 Desventajas

Teniendo en cuenta lo anteriormente comentado se encontraron las siguientes desventajas:

- Se requiere rootear el teléfono para poder usar la aplicación.
- Se requiere cambiar el firmware original por la ROM alternativa CyanogenMod.
- Es solo compatible con el Samgun Galax S1.
- El código es propietario.
- El modo monitor inhabilita el uso de la placa wifi para el resto de las aplicaciones y para conectarse a un AP en forma normal.
- La interfaz hace uso de la consola.

2.5.5 Android Open Pwn Project(AOPP) (PWM-Phone)

El Android Open Pwn Project (AOPP) (55) es una variante del Android Open Source Project (AOSP) (56). El proyecto proporciona una ROM para Android que contiene una suite de aplicaciones para realizar actividades de hacking, monitoreo y seguridad. Entre sus funcionalidades se destacan:

- Escaneo en tiempo real de redes Bluetooth e inalámbricas.
- Capacidad para correr la última versión de Kali Linux (57)(Rolling Edition) que provee el entorno para ejecutar aplicaciones de seguridad y Pentest.

2.5.5.1 Reseña histórica

El Android Open Pwn Project (AOPP) (55) es el desarrollo libre de los creadores del PwnPhone (58). El mismo es una solución comercial compuesta de un smartphone con una ROM personalizada la cual permite ejecutar las aplicaciones de Linux mayormente utilizadas en el ámbito de hacking, monitoreo y seguridad.

Nació en el 2011 con su primera versión que corría en un Nokia N900 (59), que al igual que la_solución mencionada al comienzo de este análisis (Kismet para Nokia N900), no tuvo mucha difusión. Su segunda versión llegó en Abril del 2013, con una ROM basada en una combinación de Android v4.1.2 y Kali Linux, la cual fue adaptada para la tablet Google Nexus 7 (60). A diferencia de su anterior versión, ésta utilizaba placas inalámbricas USB externas para

obtener el modo monitor mediante drivers alternativos que eran cargados en el kernel (61).

Posteriormente siguieron sacando nuevas versiones, las cuales incluyen más dispositivos soportados (62), entre ellos Google Nexus 4 (63) y Google Nexus 5 (64). En Junio del 2016 liberaron el proyecto Android Open Pwn Project (AOPP) con la finalidad de brindar a la comunidad un desarrollo que pueda ser portado a otros dispositivos.

Desde sus inicios compite con otro proyecto que persigue el mismo objetivo llamado Kali Linux NetHunter (65). Ambos están en constante actualización y poseen comunidades activas.

Android Open Pwn Project (AOPP) tiene como objetivo portar el entorno de auditoría de Kali Linux a los dispositivos móviles. Esto representa una ventaja para los usuarios que están familiarizados con la distribución. No obstante genera incomodidad manejar aplicaciones de consola desde la pantalla de un dispositivo móvil.

2.5.5.2 Análisis técnico funcional

A continuación se detallarán aspectos relacionados con la instalación y uso de la aplicación mencionada obtenidos a partir de las pruebas propias realizadas en un Google Nexus 4. Se han tomado como referencia la información proporcionada por la página oficial de la aplicación, tutoriales y videos de internet.

2.5.5.3 Procedimiento de instalación

Para instalar Android Open Pwn Project hay que rootear el teléfono e instalar la ROM alternativa brindada por el proyecto. El procedimiento es similar al explicado para la Aplicación BcMon, solo que en vez de instalar CyanogenMod se instala la ROM del proyecto AOPP, que ya tiene incorporada toda la suite de aplicaciones de Kali Linux.

2.5.5.4 Forma de uso

Una vez iniciada la ROM alternativa, la misma contiene accesos directos a las aplicaciones de consola que se encuentran en la distribución de Kali Linux. La forma de uso de las aplicaciones es equivalente a usarlas desde una Pc, donde el método de entrada es por teclado.

Algunos usuarios aprovechan la ventaja de tener toda la funcionalidad de Kali Linux en un celular para reemplazar una computadora de escritorio haciendo streaming (66) a una pantalla más grande y utilizando un teclado físico externo.

Otros usuarios lo utilizan para poder dejar el celular anclado en un lugar estratégico que quieren auditar y lo manejan en forma remota por SSH aprovechando la conexión telefónica.

2.5.5.5 Resultados de la ejecución y funcionalidad

Se llevaron a cabo pruebas equivalentes a las permitidas por las anteriores aplicaciones, y como resultado se obtuvo que la interfaz de consola no es la indicada cuando se quiere hacer auditorías en las que se necesita operar desde el celular para recorrer espacios físicos.

En las figuras 2.10 y 2.11 se puede apreciar su interfaz de consola

Fig. 2.10 – Android Open Pwn Project (AOPP)

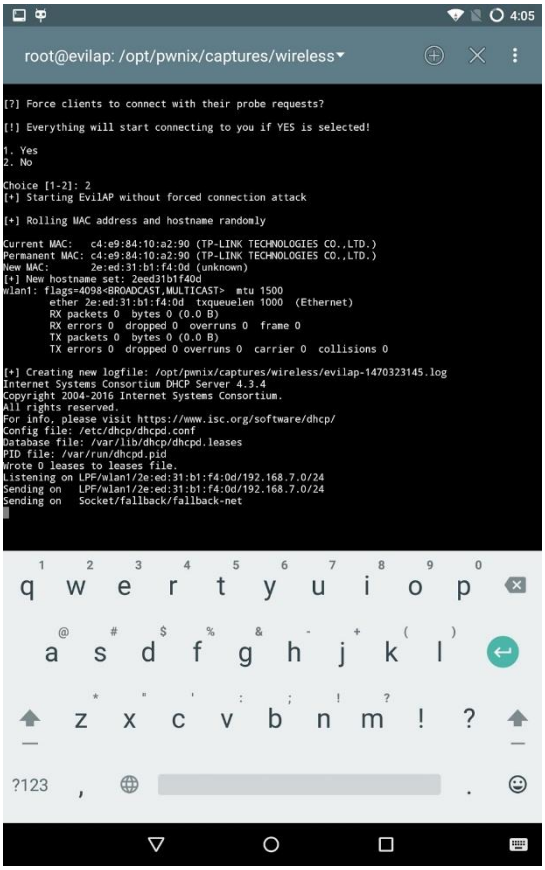


Fig. 2.11 - Android Open Pwn Project (AOPP)

2.5.5.6 Desventajas

Teniendo en cuenta lo anteriormente comentado se encontraron las siguientes desventajas:

- Se requiere rootear el teléfono para poder usar la aplicación.
- Se requiere cambiar el firmware original por la ROM alternativa.
- Oficialmente es compatible solo con algunos dispositivos: Google Nexus 5, Nexus 7 y Nexus 9 y Nvidia Shield tablet.
- Solo es compatible con algunas placas inalámbricas USB.
- La aplicación no permite hacer un ataque de desautenticación para poder obtener los handshake WPA2.
- La interfaz de consola no resulta práctica para trabajar en contextos donde se necesita portabilidad.

2.6 Cuadro comparativo

A continuación se detalla un cuadro comparativo que permite visualizar de forma sencilla las principales características de las soluciones analizadas.

	KisMet p Nokia N900	**BcMon**	**Android Pcap**	**PwnAir Pro**	**AOPP**
Fecha de lanzamiento	abril 2010	septiembre 2012	diciembre 2012	mayo 2014	junio 2016
Sistema Operativo Compatible	Maemo	Android	Android	Android	Android
Tipo de placa de red inalámbrica utilizada	Interna	Interna	Externa	Interna	Externa
Método de Obtención Modo Monitor	Root + Firmware Modificado	Root + Firmware Modificado	Driver en Espacio de Usuario	Root + Firmware Modificado	Root + Firmware Modificado
Complejidad de la Instalación	Moderada	Alta	Baja	Alta	Alta
Riesgo de la instalación	Alto	Alto	Ninguno	Alto	Alto

Cantidad de Teléfonos Compatibles	1	4	>1	1	4
Cantidad de chipset de red compatibles	1	2	1	1	<5
Tipo de interfaz	Consola	Consola	Menus + Botones	Híbrida: Menús/botones + Consola	Híbrida: Menús/botones + Consola
Grado de usabilidad de la interface	Bajo	Bajo	Alto	Moderado	Bajo
Consumo de batería	Alto	Alto	Alto	Alto	Alto

Tabla 2.1 – Cuadro comparativo de aplicaciones disponibles

2.7 Problemas actuales

Se han identificado los siguientes problemas en las alternativas analizadas:

- Rooteo y cambio de firmware/ROM para obtener el modo monitor
- Consumo de la placa inalámbrica y la duración de la batería
- Suite de herramientas y su interfaz de consola

En lo que resta del capítulo se explicará por qué se considera que cada uno de estos puntos representa una problemática en las soluciones analizadas.

2.7.1 Rooteo y cambio de firmware/ROM para obtener el modo monitor

De las 5 soluciones analizadas, 4 de ellas, incluyendo la más actual, requieren rootear el dispositivo y cambiar la ROM original por una alternativa para obtener el modo monitor de la placa inalámbrica.

La ROM alternativa es necesaria porque contiene principalmente un driver modificado que habilita el modo monitor, el cual es necesario para que la placa inalámbrica pueda escuchar todo el tráfico de la red. Sin este modo, la placa inalámbrica solo escucha lo que está destinado a ella, limitando las tareas de auditoría.

Cambiar la ROM original de un dispositivo Android por una ROM alternativa es similar a cambiar de sistema operativo en una PC, aunque presenta algunas diferencias en la forma de instalación, el hardware que soporta y los riesgos que conlleva.

Por lo general, las PCs están pensadas para que el usuario final pueda cambiar el sistema operativo. Teniendo en cuenta esto, las distribuciones de Linux de propósito general, como es el caso de Ubuntu, se enfocan en tratar de que su distribución se pueda instalar en cualquier PC. Esto requiere soportar la mayor cantidad de hardware posible y facilitar su instalación mediante herramientas automatizadas que brindan al usuario comodidad para hacerlo en poco tiempo y sin complicaciones.

Que un dispositivo de hardware esté soportado o no, depende principalmente de que exista un driver para el mismo y que esté contenido o pueda ser cargado en el Kernel de Linux. Normalmente los fabricantes de componentes de PC facilitan sus drivers para que estos puedan ser incluidos en los sistemas operativos a fin de garantizar e incentivar el uso de sus componentes.

En el caso Android, la situación es distinta. Android Open Source Project (AOSP) se encarga de mantener, desarrollar y liberar las versiones del sistema operativo pero no de incorporar los drivers necesarios para dar soporte a los diferentes dispositivos móviles. Esta tarea queda a cargo de los fabricantes, quienes desarrollan los drivers correspondientes para garantizar el buen funcionamiento del hardware fabricado. Estos drivers normalmente son privativos y están empaquetados en la ROM stock. Hay que tener en cuenta que por lo general los fabricantes no desean que los usuarios finales puedan crear e instalar ROMs alternativas en sus dispositivos, por lo que no liberan sus drivers ni documentación relacionada como para que sean desarrollados por otros. Esto

dificulta el proceso de desarrollo e instalación de una ROM alternativa, provocando que muchos dispositivos no sean soportados.

Teniendo en cuenta lo anteriormente mencionado, cambiar la ROM de un dispositivo Android no es sencillo e implica los siguientes riesgos (67):

1. Pérdida de la garantía: El primer y más importante punto a aclarar es que realizando el proceso de rooteo en el terminal e instalando las aplicaciones necesarias, se anula la garantía del fabricante. Sin embargo este proceso en casi todos los terminales es reversible.
2. Hay que desbloquear el Bootloader: Para poder cambiar la ROM stock, hay que desbloquear el bootloader. Este es el encargado de verificar que versión de Android se está utilizando y, en caso de que no sea la especificada por el vendedor, bloqueará el inicio del móvil.
3. Se anulan las actualizaciones vía OTA: Al hacer root se está modificando el sistema, por lo que en los pasos previos a una actualización OTA se detectará que el sistema ha sido modificado, inhabilitando el servicio de actualización oficial.
4. Disminuye la seguridad del sistema: Este punto quizá es el más importante, ya que el fin principal de hacer root a un terminal es poder dar permisos a determinadas aplicaciones para que realicen cambios en el sistema. El root aumenta los riesgos de la seguridad de nuestros datos. Tener permisos de administrador es equivalente a tener acceso completo a nuestro teléfono. Este privilegio no es sólo nuestro, sino también de las aplicaciones que se instalan y, si son peligrosas o están infectadas, pueden tener luz verde para obtener la información que deseen. Aplicaciones de gestión de permisos como SuperSU mostrarán un aviso cada vez que uno de estos programas intente hacer uso del root, pero la prudencia siempre es el mejor consejero y comprobar las aplicaciones instaladas o que se van a instalar, siempre es buena idea (68). De ahí que este proceso esté destinado a usuarios más avanzados, porque con una modificación del sistema incorrecta podemos dejar el smartphone corrupto.
5. Puede crear conflicto con algunas aplicaciones: Como cualquier modificación de software, este proceso puede entrar en conflicto con otras aplicaciones. Un ejemplo claro es la aplicación WhatsApp que informa al instalarla de que se está utilizando una versión modificada del sistema y

esto podría conllevar algún tipo de incompatibilidad en alguna función de la aplicación.

6. El dispositivo puede quedar inutilizable: debido a una falla en el proceso del cambio de ROM o por la instalación de una ROM incompatible con el dispositivo, este puede quedar inutilizable.
7. Inestabilidad del sistema: El dispositivo puede presentar mal funcionamiento causado por la falta o incompatibilidad de los drivers alternativos que posee la ROM.
8. Explotación de vulnerabilidades: Para obtener permisos de superusuario se utilizan aplicaciones de terceros, que obtienen el acceso root mediante la explotación de una vulnerabilidad. Esto atenta contra la seguridad del dispositivo.
9. Se pueden perder los datos del dispositivo: rootear y cambiar la ROM suele requerir resetear el teléfono a valores de fábrica antes de realizar el procedimiento. Este reseteo implica borrar todo el contenido del dispositivo, incluyendo datos del usuario. Por lo tanto, rootear y modificar la ROM original posee un riesgo, requiere de conocimientos avanzados y no es aplicable a todos los dispositivos.

2.7.2 Consumo de la placa inalámbrica y la duración de la batería

En todas las soluciones, la placa inalámbrica es el elemento principal de trabajo. Algunas han optado por utilizar la placa interna del dispositivo. Como consecuencia directa, solo queda habilitada para las aplicaciones de auditoría, provocando el mal funcionamiento del gestor de energía que conlleva a un rápido agotamiento de la batería.

Otras de las soluciones han optado por utilizar placas inalámbricas externas, las cuales tienen la ventaja de no afectar el normal funcionamiento de la placa interna, pero su alto consumo eléctrico posee dos problemas:

- Acorta la duración de la batería.
- En algunos dispositivos, el puerto USB no es capaz de entregar la corriente necesaria para el correcto funcionamiento de la placa inalámbrica externa.

La batería es una pieza fundamental para garantizar la portabilidad de los Smartphone. De ella depende que un dispositivo pueda tener la energía eléctrica necesaria para su funcionamiento en un período de tiempo. Lamentablemente las baterías no han evolucionado tanto como el resto de los componentes, siendo actualmente uno de los limitantes de las tecnologías móviles. Resulta importante hacer una correcta administración de la misma y utilizar dispositivos internos y externos de bajo consumo.

2.7.3 Suite de herramientas y su interfaz de consola

Las soluciones analizadas han incorporado como herramientas de trabajo aquellas aplicaciones de PC que son de uso habitual en las auditorías de redes inalámbricas. Para hacerlo se han enfocado en darle portabilidad a la mayor cantidad de aplicaciones ya existentes, resignando aspectos de usabilidad y eficiencia.

Por un lado, para los desarrolladores y usuarios, representa una ventaja portar una herramienta con la que se está familiarizado, que evita tener que aprender a utilizarla. Pero por otro lado representa dos grandes desventajas:

- Las herramientas portadas mantienen su interfaz de consola, la cual no resulta práctica para utilizarla desde un dispositivo móvil. Las interfaces de consola están pensadas y optimizadas para ser usadas desde una PC, donde se dispone de una pantalla más grande para su visualización y un teclado físico que permite una rápida escritura y atajos. Un dispositivo móvil carece de estas características porque está pensado para optimizar la portabilidad física del mismo. A diferencia de las PC, el método de entrada principal de un smartphone es la pantalla táctil y la utilización de uno o dos dedos, por lo que las interfaces gráficas se caracterizan por ser simples y ágiles mediante la utilización de botones y menús navegables.
- Las herramientas de PC portadas normalmente no son diseñadas pensando en la eficiencia de recursos de hardware. Portar estas herramientas a un dispositivo móvil, sin tener en cuenta aspectos como la cantidad de procesamiento que realizan, puede traer problemas de sobrecalentamiento del dispositivo y una corta duración de la batería.

Habiendo dicho lo anterior, para portar una herramienta a Android es importante primero hacer una selección de aquellas que hagan un uso adecuado de los recursos de hardware disponibles en un dispositivo móvil, y adaptar su interfaz gráfica para que sea práctica de usar.

2.8 Conclusión

A lo largo de este capítulo se pudo explicar en qué consiste la etapa de relevamiento de un Pentest de redes inalámbricas y cuáles son las tareas asociadas. Luego se presentaron las aplicaciones existentes que permiten realizarlas y las problemáticas asociadas a su uso e instalación. Queda en evidencia que ninguna de las aplicaciones representa una solución sencilla y práctica para poder llevar a cabo las tareas mencionadas, lo cual respalda la decisión de implementar una aplicación que resuelva las problemáticas planteadas.

Capítulo 3

3. Solución Propuesta

Este capítulo tiene como finalidad presentar las soluciones a las problemáticas mencionadas en el capítulo anterior: rooteo y cambio de firmware/ROM para obtener el modo monitor; consumo de la placa inalámbrica y duración de la batería; suite de herramientas y su interfaz de consola.

Posteriormente para desarrollar las soluciones presentadas, se diseñó la arquitectura de un sistema que posee los siguientes componentes: una placa de red inalámbrica, un driver Java, una API Android USB Host, una suite de herramientas y la aplicación Pentest Security App. En conjunto con este diseño, se implementó la aplicación mencionada la cual combina la API Android USB Host con el driver Java y la suite de herramientas, para comunicarse y hacer uso de la placa de red inalámbrica. Para finalizar, se describe el entorno de desarrollo y las librerías que se utilizaron en la implementación de la aplicación móvil.

3.1 Problemáticas y su solución

En el capítulo anterior se plantearon tres problemáticas importantes que deben solucionarse para lograr el desarrollo de una herramienta práctica y funcional, que sirva para llevar a cabo la etapa de relevamiento de un Pentest de redes inalámbricas.

Teniendo en cuenta lo anterior, se abordará la solución de cada una de las problemáticas planteadas, las cuales posteriormente serán reunidas e integradas para desarrollar la solución principal.

Las problemáticas expresadas en el capítulo anterior son:

- Rooteo y cambio de firmware/ROM para obtener el modo monitor
- Consumo de la placa inalámbrica y duración de la batería
- Suite de herramientas y su interfaz de consola

3.1.1 Rooteo y cambio de firmware/ROM para obtener el modo monitor

Como se mencionó en el Capítulo 2, el modo monitor de la placa inalámbrica es un recurso valioso a la hora de hacer un Pentest, pero obtener dicho modo a través del rooteo y/o cambio del firmware original del dispositivo, presenta complicaciones, riesgos de seguridad y disminuye la cantidad de dispositivos soportados.

Se necesita entonces una solución que no precise de los requerimientos anteriores. Para este fin se propone utilizar la estrategia empleada por la aplicación Android Pcap (Kismet), la cual combina la API USB host de Android, con una placa inalámbrica USB externa y un driver para Linux portado a Java, logrando que la aplicación corra íntegramente en espacio de usuario, sin permisos especiales o necesidad de alterar el kernel original.

En las soluciones donde se requiere instalar una ROM alternativa, la cantidad de dispositivos compatibles está determinada por unos pocos terminales ya que no todos cuentan con la posibilidad de tener sistemas personalizados. Por el contrario, en el método elegido para el desarrollo del presente trabajo, la cantidad de dispositivos soportados es mucho mayor dado que en la actualidad existe una gran cantidad de smartphones con versión de Android 4.0 o superior y soporte de hardware para USB HOST.

3.1.2 Consumo de la placa inalámbrica y la duración de la batería

En el punto anterior se mencionó la intención de trabajar con una placa inalámbrica externa. Teniendo en cuenta la problemática explicada en el Capítulo 2 respecto del rápido agotamiento de la batería, se propone trabajar con placas inalámbricas externas de bajo consumo, cuyo chipset sea de uso popular entre los fabricantes, con la intención de asegurar un mayor número de dispositivos comerciales compatibles.

3.1.3 Suite de herramientas y su interfaz de consola

En el capítulo anterior se explicó que las soluciones analizadas hicieron foco en otorgarle portabilidad a la mayor cantidad de aplicaciones de PC existentes, resignando aspectos de usabilidad y eficiencia.

Para abordar este problema se propone empezar por reducir la dimensión de herramientas a analizar, centrándose en las funcionalidades que permiten realizar las tareas involucradas en la etapa de relevamiento del Pentest de redes inalámbricas, para luego seleccionar aquellas que hagan un uso adecuado de los recursos de hardware disponibles en un dispositivo móvil. Luego, se integrarán en una interfaz gráfica que facilite su uso.

3.2 Arquitectura del Sistema

Para desarrollar las soluciones presentadas anteriormente, se diseñó la arquitectura de un sistema que posee los siguientes componentes: una placa de red inalámbrica, un driver JAVA, una API Android USB Host, una suite de herramientas y la aplicación Pentest Security App. En conjunto con este diseño se implementó la aplicación mencionada, la cual combina la API Android USB Host con el driver Java y la suite de herramientas, para comunicarse y hacer uso de la placa de red inalámbrica.

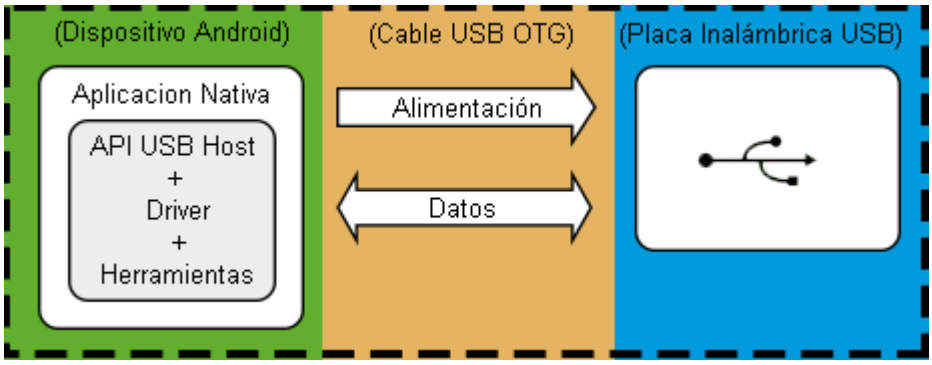


Fig. 3.1 – Componentes que intervienen en la solución

3.2.1 Placa de red inalámbrica

La placa de red inalámbrica USB es el periférico con el que se comunicará la aplicación para poder realizar las tareas de auditoría. Para elegir la placa inalámbrica adecuada hay que considerar los siguientes requisitos:

- Driver en modo usuario: Es importante que la placa inalámbrica cuente con su driver portado a Java, para poder ser utilizado en espacio de usuario, o en su defecto, que posea un driver de Linux open source para poder portar manualmente el código.
- Bajo consumo: Teniendo en cuenta que el consumo de las placas inalámbricas puede provocar un rápido agotamiento de la batería, la placa elegida debe poseer un bajo consumo eléctrico.
- Tamaño reducido: Para asegurar y acompañar la portabilidad de los dispositivos móviles, es importante tener en cuenta que las dimensiones físicas de la placa inalámbrica deben ser las mínimas posibles. Esto ayudará a que la integración física del periférico no comprometa la usabilidad de la solución planteada.

Teniendo en cuenta lo anterior se eligió la familia de placas inalámbricas con chipsets RTL8192CU/RTL8188CUS por:

- La amplia variedad de opciones comerciales: Estos chipsets han sido incorporados por muchas marcas dotando al mercado de una amplia variedad de dispositivos, lo cual facilita la tarea de adquisición/compra. En la actualidad se han podido detectar 116 placas USB compatibles, de las cuales 69 tienen chipset RTL8188CUS (69) y 47 tienen chipset RTL8192CU (70).
- El bajo consumo y reducido tamaño: estos chipset son muy populares por ser usados en dongles inalámbricos USB de muy bajo consumo (alrededor de 40 mA) y de reducido tamaño (aproximadamente 1.5 cm2)
- Disponibilidad de driver portado a Java: Existe un driver de código libre desarrollado por Milen Rangelov (71).

3.2.2 Driver Java

Un driver o controlador de dispositivo (en inglés: device driver) es un programa informático que permite interactuar con un periférico, haciendo una abstracción del hardware y proporcionando una interfaz (posiblemente estandarizada) para utilizar el dispositivo. Es una pieza esencial del software, sin la cual el hardware sería inutilizable.

Diseñar un driver, que se pueda comunicar con la placa inalámbrica USB externa, habilitar el uso del modo monitor y correr en el espacio de usuario de una aplicación de Android, implica que el mismo debe estar escrito en Java y debe interactuar con la API USB Host. Esta permite comunicarse con el dispositivo externo a nivel de bytes, es decir una comunicación en modo Raw, y quedará a cargo del driver interpretar esos bytes.

El modo monitor permite a las placas inalámbricas poder escuchar todo el tráfico que circula por la red y en algunos casos, también hacer inyección de paquetes (emisión de paquetes modificados).

El driver elegido, desarrollado por Milen Rangelov (71), permite la utilización del modo monitor con inyección de paquetes. Este es necesario para el correcto funcionamiento de la suite de herramientas que serán seleccionadas posteriormente. Originalmente soportaba solo 4 placas USB comerciales con los chipsets RTL8192CU y RTL8188CUS, lo cual fue modificado para llegar a un total de 60. No se pudo soportar las 116 opciones comerciales disponibles en la actualidad, debido a la falta de los datos "Vendor ID" y "Device ID" de algunas de ellas.

3.2.2.1 Dispositivos inalámbricos soportados

A continuación se presenta una tabla con los dispositivos comerciales soportados:

Nombre	Formato Físico	Vendor ID	Device ID	Chipset
Netgear WNA1000Mv1	micro dongle	846	9041	RTL8188CUS
On Networks N150MA	nano dongle	846	9042	RTL8188CUS
D-Link DWA-121 rev A1	nano dongle	2001	3308	RTL8188CUS
Planex GW-USNano2	nano dongle	2019	ab2a	RTL8188CUS
Planex GW-USValue-EZ	nano dongle	2019	ed17	RTL8188CUS
Planex GW-USWExtreme	nano dongle	2019	ed17	RTL8188CUS
Edimax EW-7811Un	nano dongle	7392	7811	RTL8188CUS
ISY IWL 2000	nano dongle	050d	11f2	RTL8188CUS
Belkin F7D1102	nano dongle	050d	1102	RTL8188CUS
Hercules HWNUp-150	nano dongle	06f8	e033	RTL8188CUS
ASUS USB-N10 Nano	nano dongle	0b05	17ba	RTL8188CUS
Airlink101 AWLL5088	nano dongle	0bda	8176	RTL8188CUS
CC&C WL-6200-V1	micro dongle	0bda	8176	RTL8188CUS
CNet CQU-906	nano dongle	0bda	8176	RTL8188CUS
EDUP EP-N8508	nano dongle	0bda	8176	RTL8188CUS
EDUP EP-N8508GS	nano dongle	0bda	8176	RTL8188CUS
Encore ENUWI-1X42	micro dongle	0bda	8176	RTL8188CUS
Encore ENUWI-1X45	micro dongle	0bda	8176	RTL8188CUS

Encore ENUWI-1XN42	dongle	0bda	8176	RTL8188CUS
Encore ENUWI-1XN45	dongle	0bda	8176	RTL8188CUS
Rosewill RNX-MiniN1	nano dongle	0bda	8176	RTL8188CUS
Airlink101 AWLL5099	nano dongle	0bda	8176	RTL8188CUS
B-LINK BL-LW05-5R	nano dongle	0bda	8176	RTL8188CUS
Netis WF-2120	nano dongle	0bda	8176	RTL8188CUS
TP-LINK TL-WN723N v2	micro dongle	0bda	8176	RTL8188CUS
Diamond WL700RN rev 2	dongle	0bda	8176	RTL8188CUS
Encore ENUWI-1XN4M	nano dongle	0bda	8176	RTL8188CUS
FAST FW150UM	mini dongle	0bda	8176	RTL8188CUS
ACELINK WU110EC	nano dongle	0bda	8176	RTL8188CUS
Monoprice 8072	nano dongle	0bda	8176	RTL8188CUS
Sabrent USB-A11N	dongle	0bda	8176	RTL8188CUS
SkyCity SY-W8509	nano dongle	0bda	8176	RTL8188CUS
TP-LINK TL-WN725N v1	nano dongle	0bda	8176	RTL8188CUS
TRENDnet TEW-648UBM	nano dongle	20f4	648b	RTL8188CUS
Hawking HWDN3	dongle	0	19	RTL8192CU
Hawking HWUN4	micro dongle	0	20	RTL8192CU
ZyXEL NWD2205	micro dongle	586	341f	RTL8192CU
Netgear WNA3100M	micro dongle	846	9021	RTL8192CU
On Networks N300MA	micro dongle	846	f001	RTL8192CU
D-Link DWA-135	dongle	2001	3309	RTL8192CU
D-Link DWA-131 rev B1	micro dongle	2001	330d	RTL8192CU
D-Link DWA-133	dongle	2001	330a	RTL8192CU
Planex GW-USEco300	micro dongle	2019	ab2b	RTL8192CU
TP-LINK TL-WN8200ND	dongle	2357	100	RTL8192CU
Edimax EW-7822GTN	micro dongle	7392	7822	RTL8192CU
Belkin F9L1004	micro dongle	050d	1004	RTL8192CU
ISY IWL 4000	micro dongle	050d	21f2	RTL8192CU
Belkin F7D2102	micro dongle	050d	2103	RTL8192CU
ASUS USB-N13 rev B1	dongle	0b05	17ab	RTL8192CU
Encore ENUWI-2XN42	dongle	0bda	8178	RTL8192CU
Encore ENUWI-2XN45	dongle	0bda	8178	RTL8192CU
Rosewill RNX-N250UBE	dongle	0bda	8178	RTL8192CU
TP-LINK TL-WN821N v4	dongle	0bda	8178	RTL8192CU
TP-LINK TL-WN823N v1	micro dongle	0bda	8178	RTL8192CU
TP-LINK TL-WN822N v3	dongle	0bda	8178	RTL8192CU
Manhattan 525527	micro dongle	0bda	8178	RTL8192CU
PEARL 300 MBit	micro dongle	0bda	8178	RTL8192CU
Sitecom WLA-2102	micro dongle	0df6	70	RTL8192CU
Sitecom WLA-4001	micro dongle	0df6	61	RTL8192CU
TRENDnet TEW-624UB rev D1	micro dongle	20f4	624d	RTL8192CU

Tabla 3.1 – Listado de dispositivos soportados por la aplicación

3.2.3 Android API USB Host

Android 3.1 introdujo el soporte para USB On-The-Go (OTG). Esto permite que un dispositivo Android actúe como host USB (como un ordenador de sobremesa) en lugar de como un periférico USB Por ejempo, un mouse,

como medio de entrada, para interactuar en la pantalla del smartphone. Sin embargo, muchos periféricos no tienen soporte, por parte de Android, con un controlador nativo. En otras palabras, queda a cargo de la aplicación crear compatibilidad mediante un driver para ese dispositivo.

En la figura 3.2 se puede observar cómo quedan agrupados los diferentes componentes de software en el entorno de Android.

Fig. 3.2 – Aquitectura de la plataforma Android

La forma de conseguir que un periférico funcione con Android, cuando no hay soporte nativo, es utilizar la API USB Host para implementar el driver del periférico en espacio de usuario. Dicha API permite que las aplicaciones se registren para ser notificadas cuando un periférico específico es conectado. Esto se convierte en el punto de entrada del controlador personalizado. A partir de ahí, se puede realizar E/S a nivel de byte con el periférico. Desde el punto de vista del sistema operativo, esos bytes no tienen ningún significado, lo cual ofrece libertad y flexibilidad al momento de implementar un controlador en espacio de usuario. Debido a que la API USB Host está oficialmente admitida, no es necesario rootear el dispositivo para interactuar con los periféricos. Gracias a esto, las aplicaciones que utilizan este tipo de controladores, se pueden publicar en Google Play Store.

Para poder utilizar la funcionalidad de USB Host, se deben cumplir los siguientes requisitos:

1. Que el dispositivo Android soporte en su hardware la modalidad de USB Host.
2. Que el dispositivo tenga una versión de Android 3.1 o superior.
3. Utilizar un cable USB OTG para conectar el periférico al dispositivo Android, el cual habilita el modo USB Host de este último.

3.2.4 Suite de herramientas

Teniendo en cuenta lo mencionado en el capítulo 2 sobre tareas realizadas en la etapa de relevamiento de un Pentest de redes inalámbricas se clasificó la funcionalidad a implementar en tres categorías. La primera de ellas, abarca las tareas a realizar cuando no se tiene acceso a la red: enumerar los AP encontrados (visibles y ocultos) y obtener de ellos información valiosa como el protocolo de seguridad, canal de transmisión, nivel de potencia y cantidad de Stations asociados. Esto le permitirá al auditor verificar y determinar, entre otras cosas, si hay APs de tipo rouge y si el nivel de encriptación elegido es apropiado para el tipo de red. Además se podrá identificar y enumerar las estaciones asociadas a cada AP, para luego realizarles ataques de desautenticación con el fin de obtener su handshake.

La segunda categoría comprende las tareas a realizar cuando se tiene acceso a una red: se podrá identificar la IP y MAC de los dispositivos que están conectados a la misma y será posible determinar qué servicios operan sobre cada uno mediante el análisis de puertos abiertos (UDP, TCP).

Por último, la tercera categoría engloba la funcionalidad necesaria para poder exportar los resultados: la aplicación será capaz de generar un informe detallado con la evidencia obtenida y un archivo pcap con el tráfico de red capturado. Este último podrá ser enviado a un servidor remoto para su posterior análisis.

Todas las funcionalidades mencionadas poseen las siguientes características:

- Admiten su manejo desde una interfaz nativa de Android, con la finalidad de proporcionar usabilidad y experiencia al usuario.
- Están acordes a las posibilidades y limitaciones que posee un dispositivo móvil, ya que no requieren de un gran poder de cómputo y tiempo de ejecución, lo cual influiría negativamente en la duración de la batería.

3.2.5 Aplicación Pentest Security App

Pentest Security App es una aplicación Android nativa, desarrollada en el lenguaje Java. Combina la API USB Host y el driver Java desarrollado por Milen Rangelov, con el conjunto de funcionalidades descriptas anteriormente.

3.2.5.1 Nivel de API

Para llevar a cabo la implementación de la aplicación se eligió como base la API nivel 14, la cual brinda compatibilidad con dispositivos que tengan la versión de Android 4.0 o superior.

Según los datos publicados en el portal oficial de Android (72), sobre la cantidad relativa de dispositivos que usan cada versión, el 98% de los utilizados en la actualidad son compatibles con la aplicación implementada.

3.2.5.2 Diagrama de clases

La Fig. 3.3 muestra la estructura de clases que componen a la aplicación. De este diseño se destacan dos aspectos importantes: por un lado la posibilidad de poder ampliar la familia de chipsets soportados, extendiendo la superclase UsbSource y re implementando los métodos que sean necesarios para generar un nuevo driver. Y por el otro, admitir la incorporación de nuevos métodos de descubrimiento de APs, extendiendo la superclase AbstractDiscovery.

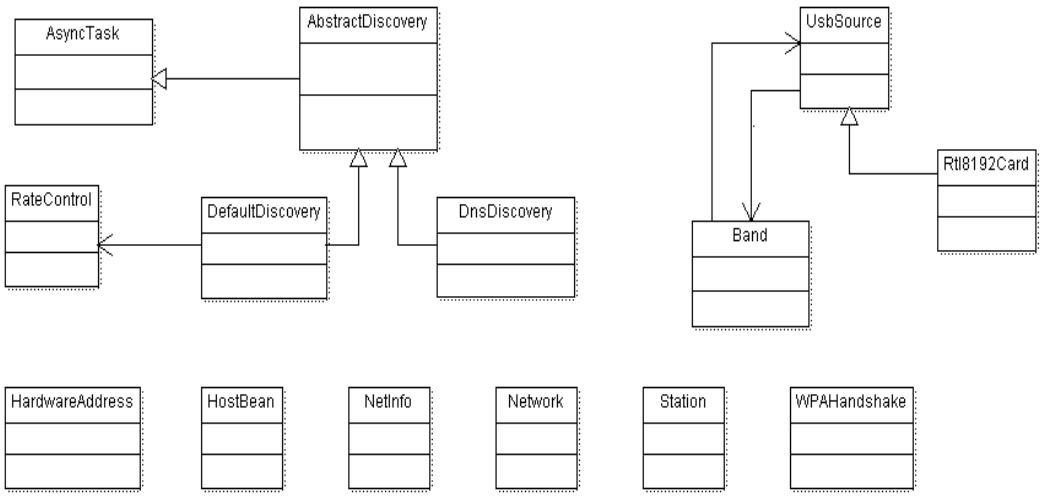


Fig. 3.3 – Estructura de clases de la aplicación

3.2.5.3 Permisos requeridos por la aplicación

Para el correcto funcionamiento de la aplicación se requieren los siguientes permisos:

- **INTERNET:** Permite abrir sockets de red para permitir el envío y recepción de datos.
- **WRITE_EXTERNAL_STORAGE:** Permite escribir en la memoria externa del dispositivo.

- **READ_EXTERNAL_STORAGE:** Permite leer de la memoria externa del dispositivo.
- **ACCESS_WIFI_STATE:** Permite acceder a la información sobre las redes inalámbricas detectadas.
- **CHANGE_WIFI_STATE:** Permite cambiar el estado de conectividad WIFI del dispositivo.
- **READ_PHONE_STATE:** Permite tener acceso, en modo lectura, al estado del teléfono y a su identidad.
- **ACCESS_NETWORK_STATE:** Permite conocer el estado de la red.

3.2.5.4 Licencias

Todos los componentes de la aplicación de software implementada han sido liberados bajo la licencia GPL versión 3, por lo cual el código genera un aporte a la comunidad de usuarios de software libre, permitiendo que se use, estudie, modifique y distribuya libremente.

3.2.5.5 Repositorio

El código fuente de la aplicación implementada se encuentra alojado en el repositorio público con dirección https:

```
https://gitlab.com/zube/pentestsecurityapp
```

3.3 El entorno de desarrollo

Como entorno de desarrollo se utilizó Android Studio (73) (74) y un conjunto de librerías. A continuación se describirán los mismos.

3.3.1 Android Studio

Se eligió Android Studio por ser el entorno de desarrollo oficial del proyecto Android, lo cual asegura la compatibilidad con las funciones nativas, como así también las actualizaciones constantes y una comunidad de desarrolladores activa.

Está principalmente basado en el software IntelliJ IDEA de JetBrains y es totalmente gratuito por medio de la licencia Apache 2.0. Otro de los beneficios

es que puede correr sobre los distintos sistemas operativos, Windows, Mac OS X y GNU Linux.

Principales Características:

- *Herramientas Lint:* Estas diversas herramientas son capaces de detectar códigos no compatibles y de arquitecturas diferentes, así mismo detecta problemas de rendimiento, usabilidad y semejanza de versiones.
- *Pro Guard:* Esta herramienta es muy útil para las aplicaciones de tipo APK, que son diseñadas para Smartphone con poca memoria. Pro Guard ayuda a reducir el código que se va a exportar a la aplicación, esto es de gran ayuda para los dispositivos de gama baja o que tengan problemas de memoria.
- *Gradle:* Sirve para gestionar en forma automatizada la construcción de proyectos de aplicaciones, testing, empaquetado o compilación.
- *Google Cloud Plataform:* Esta singular característica permite conectar al programador con los archivos que posea en la nube.

3.3.2 Versión de Android utilizada para el desarrollo

- Android Studio 2.2.2
  - Build #AI-145.3360264, built on October 18, 2016
  - JRE: 1.8.0_76-release-b03 amd64
  - JVM: OpenJDK 64-Bit Server VM by JetBrains s.r.o
- Gradle 2.14.1
- Android Plugin Version 2.2.2

3.3.3 Librerías utilizadas

- **HTTPclient, HTTPcore y HTTPmime:** son un conjunto de librerías que permiten crear un cliente HTTP. En la aplicación son utilizadas para poder enviar el pcap generado a un servidor externo.

  Versiones utilizadas: httpclient-4.3.1.jar, httpcore-4.3.jar, httpmime-4.3.jar.

- **ItextG:** es una librería Java de código libre para la generación y manipulación de PDF. En la aplicación se utilizó la versión itextg-5.5.8.jar para generar los informes.

3.4 Conclusión

Teniendo en cuenta que el principal objetivo de esta tesina es desarrollar una herramienta que sirva para el relevamiento de redes inalámbricas y el conjunto de problemáticas mencionadas en el capítulo 2, se presentaron las soluciones generales a las que se arribaron, para luego desarrollar un sistema que integre las mismas.

La problemática asociada al Rooteo y cambio de firmware para obtener el modo monitor, se solucionó con el uso de una placa inalámbrica USB externa, la API USB Host de Android y un driver Java en espacio de usuario.

Para la problemática relacionada al consumo de la placa inalámbrica y duración de la batería se decidió utilizar una placa de bajo consumo.

Respecto a la suite de herramientas y su interfaz de consola, se propuso desarrollar una interfaz fácil y práctica de usar para aquellas funcionalidades que son de importancia en la etapa de relevamiento de un Pentest de redes inalámbricas y que hacen un uso adecuado de los recursos de hardware disponibles en un dispositivo móvil.

Posteriormente se diseñó la arquitectura de un sistema que posee los componentes necesarios para desarrollar e integrar las soluciones planteadas.

Como resultado de la implementación se logró una aplicación con las siguientes características:

- Soporta una amplia variedad de placas inalámbricas de bajo costo y consumo.
- Combina la portabilidad y practicidad de un dispositivo móvil, con un conjunto de herramientas para realizar el relevamiento de redes inalámbricas.
- Es capaz de ser instalada y operada, sin requerir de modificaciones en el dispositivo móvil.

En el capítulo siguiente se mostrará la aplicación resultante y las pruebas realizadas.

Capítulo 4

4. Implementación y Resultados

En este capítulo se describe la instalación y uso de la aplicación Pentest Security App, implementada en el marco de la tesina, que permite realizar de forma práctica y sencilla las tareas relacionadas al relevamiento de un Pentest de redes inalámbricas.

Posteriormente se plantea un caso y entorno de pruebas para verificar y demostrar el correcto funcionamiento de la aplicación. Finalmente se presentan los resultados obtenidos en las pruebas realizadas.

4.1 Pentest Security App

Pentest Security App es una aplicación para Android, que permite realizar tareas relacionadas a la etapa de relevamiento de Pentest de redes inalámbricas. A continuación de describen aspectos relacionados con su instalación y uso.

4.1.1 Instalación

Para poder hacer uso de Pentest Security App se requiere disponer de un Smartphone con Android 4.0 o superior, un cable USB OTG y una placa inalámbrica compatible. Luego se debe bajar e instalar el archivo APK provisto en el repositorio Git mencionado anteriormente.

4.1.2 Pantalla Principal

La aplicación está construida utilizando los componentes visuales nativos de la plataforma Android. Consecuentemente la interface resultante es sencilla y fácil de entender y manejar.

Al iniciar la aplicación se puede observar, en la parte superior de la pantalla, una barra de estado (tal como muestra la figura 4.1 y 4.2) que permanecerá fija y reflejará en forma constante el estado de la placa inalámbrica externa. Se podrá visualizar si esta última pudo ser acoplada exitosamente o si

ocurrió algún error. En caso exitoso, la aplicación empezará a detectar y contabilizar los APs, Stations y Handshakes encontrados, reflejando periódicamente su resultado en la barra de estado. Es importante resaltar que si bien la aplicación estará analizando el tráfico de red para contabilizar los aspectos mencionados, el mismo no será guardado en un archivo, a menos que se presione el botón "Comenzar Captura", ubicado en la parte inferior de la pantalla. Consecuentemente, aparecerá en pantalla la ruta del archivo pcap generado. Para terminar la captura del tráfico de red se debe utilizar el botón "Detener Captura".

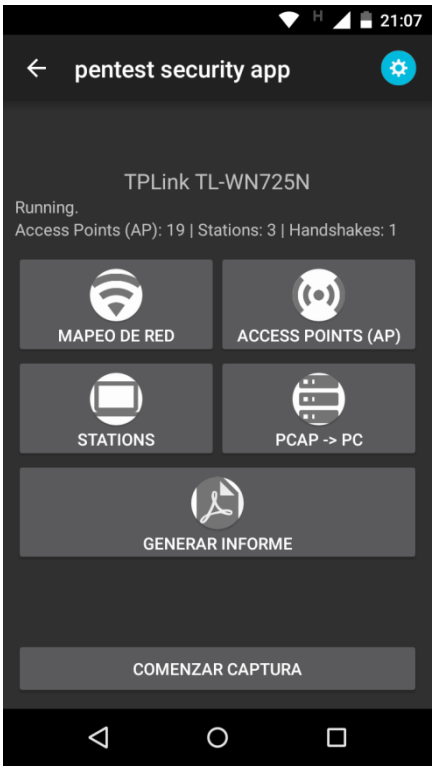


Fig. 4.1. Pantalla principal

Fig. 4.2. Pantalla principal en modo captura

En la parte central de la pantalla se encuentra un menú con 5 opciones: Mapeo de Red, Access Points (AP), Stations, Pcap ->PC y Generar Informe. Teniendo en cuenta la clasificación mencionada en la sección "Suite de herramientas" del capítulo anterior, se explica la funcionalidad de la aplicación de la siguiente manera:

- **Sin acceso a la red**: Cuando se va a relevar una red inalámbrica, es importante poder tener opciones que permitan recolectar la mayor cantidad de información sin la necesidad de tener que unirse a dicha red. Consecuentemente, en esta categoría entran las opciones "Access

Points" (AP) y "Stations", las cuales utilizan la placa inalámbrica externa en modo monitor para escuchar todo el tráfico de red.

- **Con acceso a la Red**: En esta categoría entra la opción "Mapeo de Red", que permite recolectar información de una red a la que se esté conectado, médiante la placa inalámbrica interna del teléfono.
- **Exportación de resultados:** En esta categoría entran las opciones de envió de archivos (Pcap -> PC) y Generar Informe.

4.1.3 Sin acceso a la red

La aplicación está diseñada para recolectar el tráfico que circula por la red, utilizando una placa inalámbrica externa en modo monitor, conectada al smartphone mediante un cable USB OTG. Dentro de esta categoría existen dos opciones:

4.1.3.1 Botón "Access Points (AP)"

Figura 4.3: muestra un listado de todos los puntos de acceso inalámbricos detectados, detallando el protocolo de seguridad, el canal de transmisión, la cantidad de información que recibe, el nivel de potencia y la cantidad de dispositivos conectados al mismo. Dicho listado se encuentra ordenado por protocolo de seguridad, ubicando en la parte superior a todos aquellos dispositivos que no tengan seguridad, seguidos por los WEP y por último los WPA/WPA2. A su vez, el nivel de potencia de la señal se traduce en colores y a los SSID ocultos se los etiqueta "HIDDEN" como ayuda visual para el usuario.

Referencia de color/calidad señal/nivel de potencia:

- Verde (excelente): -45 a -59 db
- Amarillo(buena): -60 a -70 db
- Naranja(regular): -71 a -76 db
- Rojo(mala): => -76 db

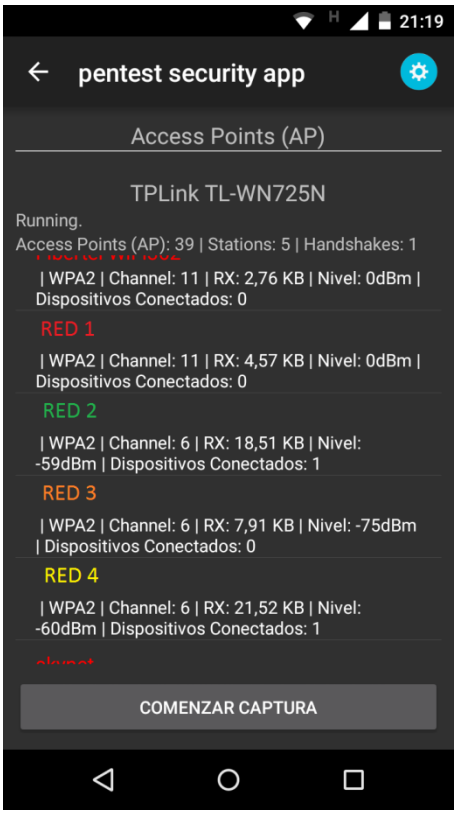


Fig. 4.3. Pantalla de Access Points (AP). Los colores son indicadores del nivel de señal de la red.

4.1.3.2 Boton "Stations"

Muestra un listado de todos los Station (Figura 4.4), describiendo su MAC, el AP al que están asociados, y la cantidad de datos enviados. Al seleccionar un Station se le puede hacer un "Ataque de Desautenticación" (Figura 4.5), el cual tiene como objetivo desconectar al Station del AP al que está asociado. A partir de este momento, si la aplicación está capturando el tráfico de la red en un pcap y el Station atacado intenta volver a conectarse al AP, se podrá obtener el handshake necesario para crackear la contraseña.

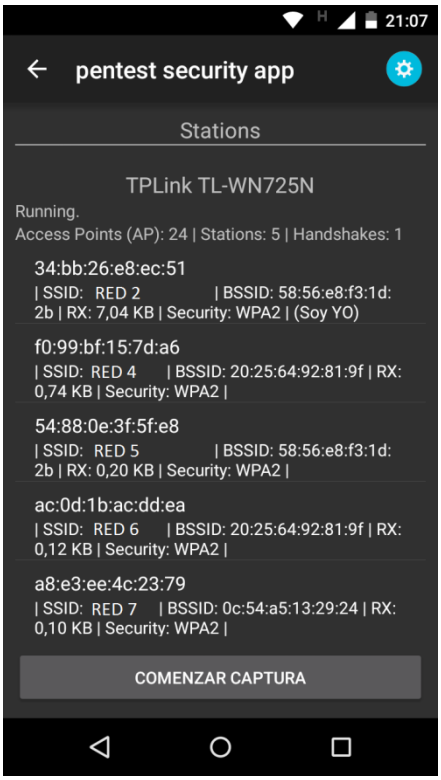


Fig. 4.4. Pantalla de Stations

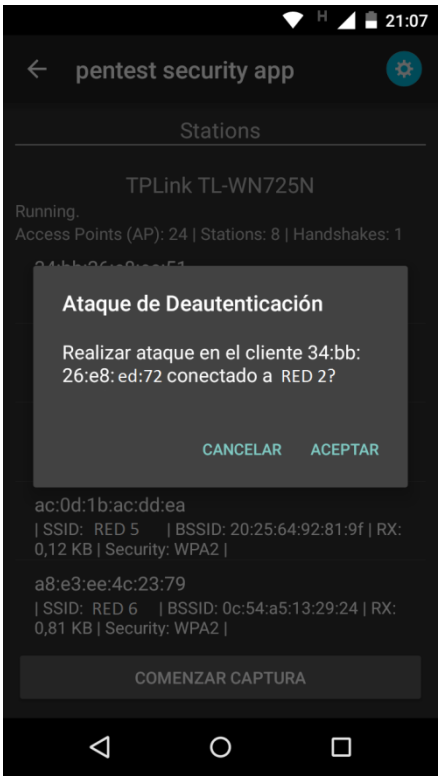


Fig. 4.5. Pantalla donde se realizar el ataque de desautenticación

4.1.4 Con acceso a la red

Una vez que se tiene acceso a la red, se puede inspeccionar la misma en búsqueda de servicios y dispositivos.

4.1.4.1 Botón "Mapeo de Red"

Este tiene como propósito brindar un detalle de los dispositivos conectados a la red inspeccionada. Esta funcionalidad utiliza la placa inalámbrica interna, en forma independiente al uso de la placa inalámbrica externa. Se listan las direcciones IP y MAC de los dispositivos encontrados (figuras 4.6 y 4.7). En la parte inferior de la pantalla se muestra una leyenda con el nombre de la red que se está relevando.

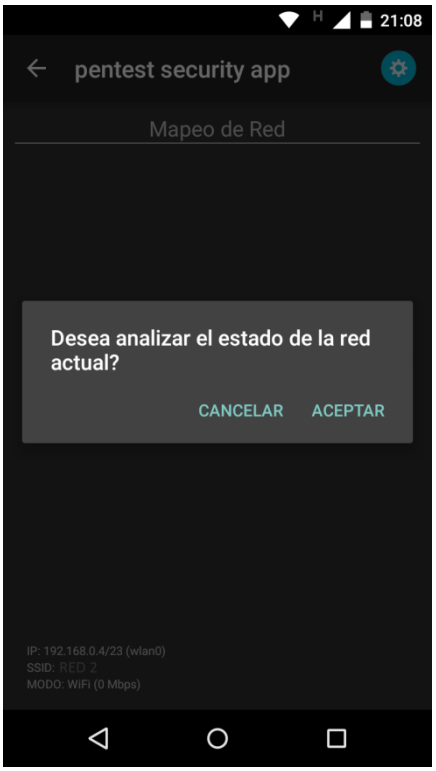


Fig. 4.6. Pantalla de Mapeo de Red.

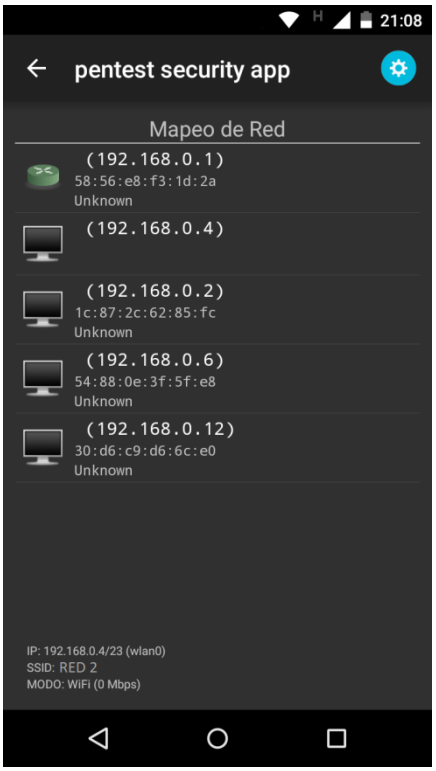


Fig. 4.7. Pantalla de dispositivos conectados en la red a la que el móvil está conectado.

Una vez finalizado el descubrimiento de los dispositivos conectados a la red, es posible seleccionar uno y realizar una búsqueda de puertos (Figura 4.8). De la misma se puede configurar el tipo de protocolo (TCP, UDP) y el número de puertos a escanear (Figura 4.9). Como resultado se obtiene un listado de los puertos abiertos encontrados, describiendo el número y protocolo de cada uno (Figura 4.10).

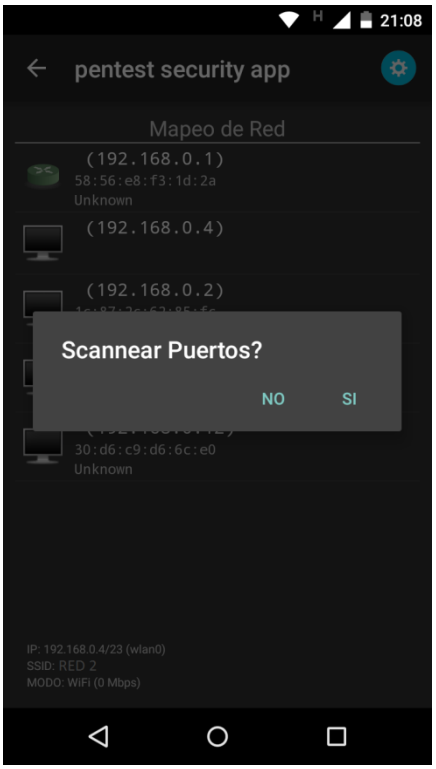


Fig. 4.8. Pantalla de Escaneo de Puertos.

Fig. 4.9. Pantalla de configuración previa al escaneo de puertos.

Fig. 4.10. Pantalla con el resultado de los puertos encontrados.

4.1.5 Herramientas de soporte:

4.1.5.1 Botón "Pcap ->PC": Envío de archivo pcap

Finalizada la etapa de relevamiento, toda la información capturada en un pcap se puede enviar a un servidor utilizando un servicio web. Para ello es necesario tener cargada la dirección IP y el puerto del webservice en la solapa de configuración de la pantalla principal (Figura 4.11). Luego se debe seleccionar un archivo del listado de los pcap (Figura 4.12).

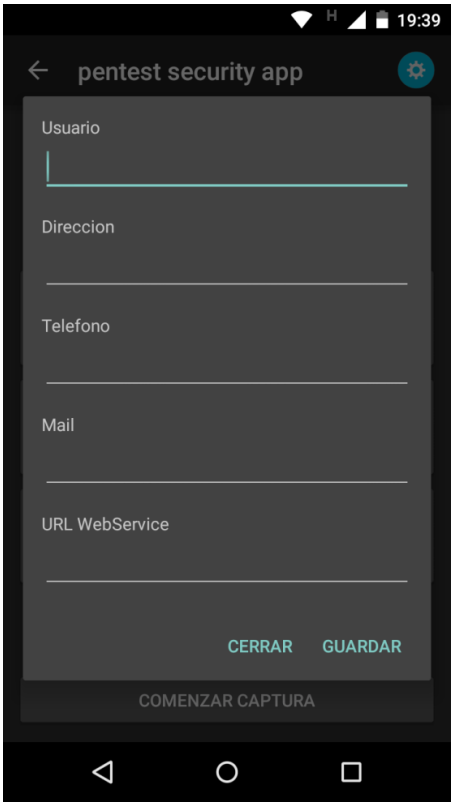


Fig.4.11 Pantalla de configuración de webserice

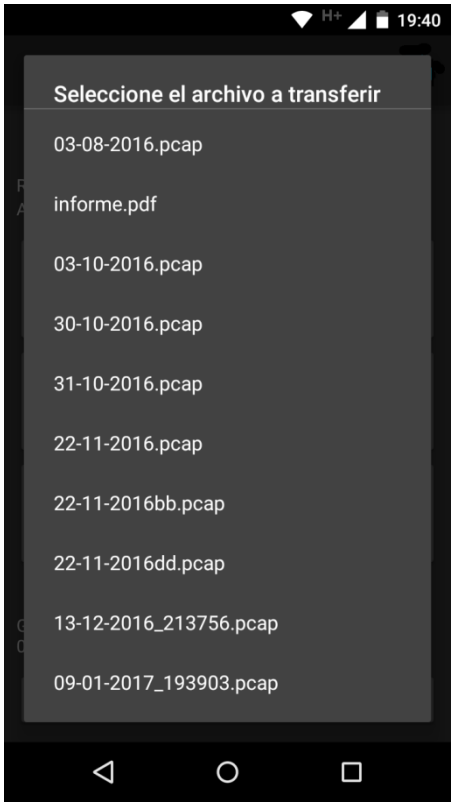
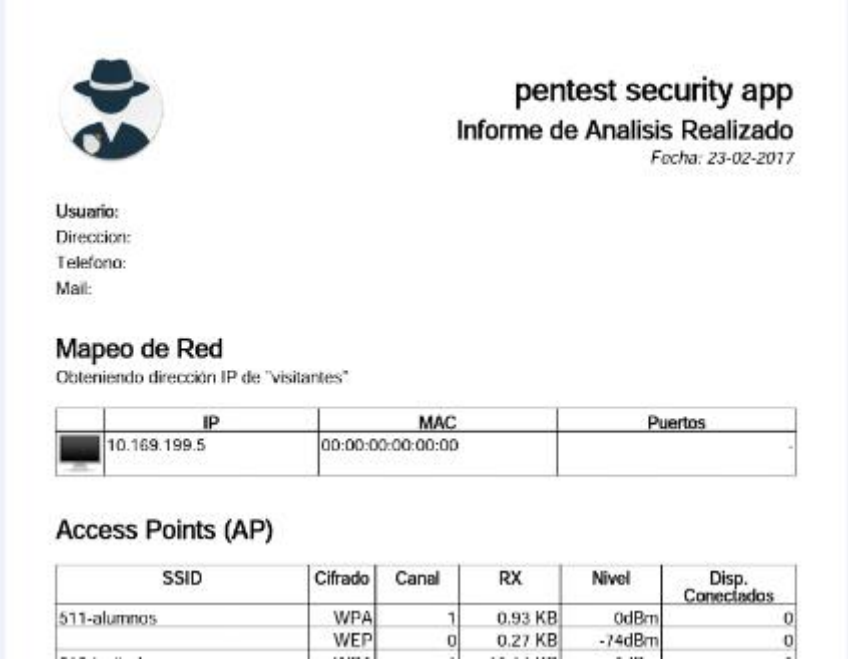


Fig. 4.12. Pantalla de Listado de Pcap del informe

4.1.5.2 Boton "Generar Informe"

La aplicación posee la funcionalidad para generar un informe con el detalle del relevamiento llevado a cabo.



pentest security app

Informe de Analisis Realizado

Fecha: 23-02-2017

Usuario:
Direccion:
Telefono:
Mail:

Mapeo de Red

Obteniendo dirección IP de "visitantes"

	IP	MAC	Puertos
	10.169.199.5	00:00:00:00:00:00	

Access Points (AP)

SSID	Cifrado	Canal	RX	Nivel	Disp. Conectados
511-alumnos	WPA	1	0.93 KB	0dBm	0
	WEP	0	0.27 KB	-74dBm	0

Fig. 4.13 – Informe generado

4.2 Entorno de Pruebas

Para poder llevar a cabo la validación de la aplicación se definió un caso de prueba y se configuró un entorno, compuesto por hardware y software. Los mismos se explican a continuación.

4.2.1 Caso de Prueba

Se releva la red con SSID "juan_router", la cual utiliza WPA2-PSK y encriptación TKIP. Se realiza una captura de paquetes durante 60 minutos, utilizando una placa inalámbrica externa, generando un pcap del tráfico capturado. Mientras se está llevando a cabo la captura, se verifica el nivel de la batería para corroborar el bajo consumo de la aplicación y de la placa inalámbrica externa. Posteriormente se envía el pcap obtenido al servidor, haciendo uso del webservice configurado, donde se lo analiza con Wireshark y se lo examina utilizando Aircrack-ng para verificar si contiene algún handshake, y luego crackearlo.

4.2.2 Componentes de hardware

4.2.2.1 Teléfono móvil

Para realizar las pruebas se utilizará un smarphone marca Motorola modelo XT1032 (comercialmente conocido como Moto G 1ra. Generación). El mismo está catalogado como un celular de gama media, lo cual permitirá evaluar si la aplicación es capaz de desempeñarse correctamente, sin requerir de mayores prestaciones. Posee una pantalla de 4.5 pulgadas, tamaño que permite buena usabilidad sin sacrificar la portabilidad. Además cuenta con un procesador ARM de 4 núcleos, 1 Gb de memoria RAM, y una batería Li-Ion de 2070 mAh. Ademas, tiene instalado la ROM original de Android 5.1.

Para llevar a cabo las pruebas y asegurar que el dispositivo no posea alguna configuración previa, que pueda alterar los resultados, se resetea el mismo a sus valores de fábrica.

Fig. 4.14 – Modelo de smartphones utilizado para las pruebas

4.2.2.2 Cable USB OTG

Se utiliza un cable USB OTG (On-The-Go) para conectar el dispositivo inalámbrico al teléfono.

Fig. 4.15 – Modelo de cable USB OTG utilizado para las pruebas

4.2.2.3 Placa de red

Para poder realizar pruebas con la aplicación se utiliza una placa inalámbrica TPLink Modelo TL-WN725N v1, la cual contiene el Chipset: RTL8188CUS. Posee un formato nano dongle haciendola ideal por su reducido tamaño. Su ID Vendor es 0bda y su Id Product es 8176.

Fig. 4.16 – Comparativa de tamaño entre placa de red y moneda

4.2.2.4 Access Point

Para la fase de pruebas se utiliza un router inalámbrico marca TP-LINK modelo TL-WR340G, al cual se le configura el SSID "juan_router", tipo de seguridad WPA2-PSK y encriptación TKIP.

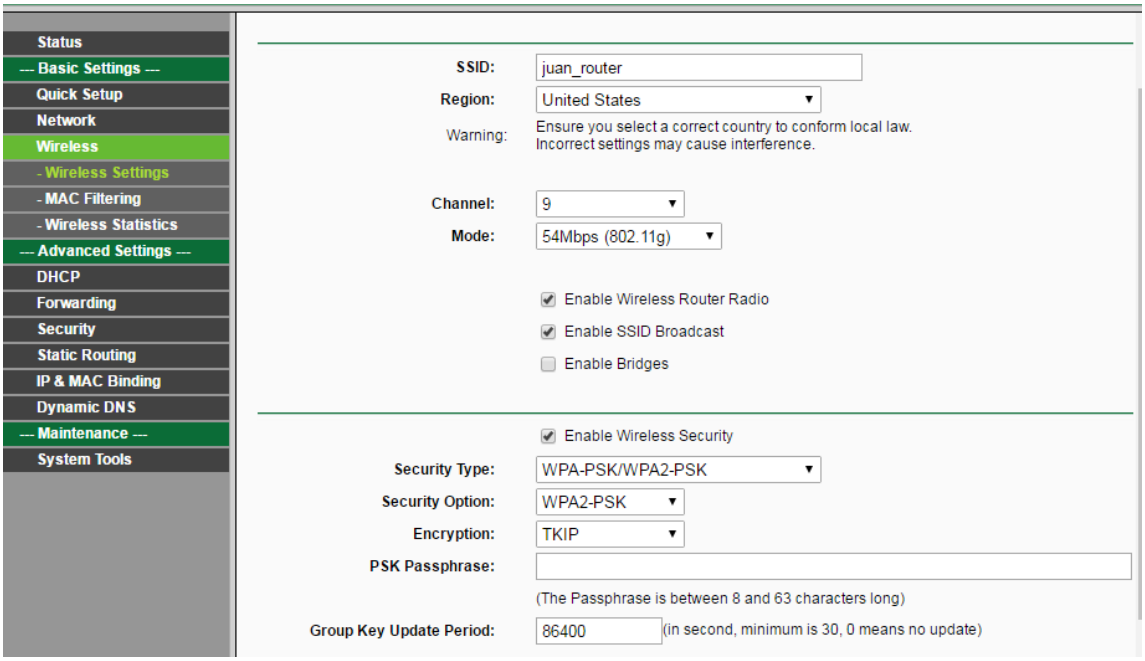


Fig. 17 – Captura de pantalla Router

4.2.3 Componentes de software

4.2.3.1 Root Checker Basic

Esta aplicación es gratuita y se puede descargar de Google Play Store. Se utiliza para corroborar si el dispositivo que la ejecuta está rooteado o no.

4.2.3.2 WebService RESTful

Como complemento, se desarrolló un servicio web para poder transferir los archivos pcap, generados por la aplicación móvil, a una PC con mayor capacidad de cómputo, para luego poder realizar tareas de análisis. Dicho desarrollo utiliza una arquitectura RESTful (75) (76) (77) que está basada en un protocolo sin estado de tipo cliente/servidor (protocolo HTTP), que cuenta con operaciones bien definidas (GET, POST, PUT, DELETE) y recursos identificados de forma única por URIs. Se implementó con un servidor Apache Tomcat (78), por ser un proyecto de código abierto basado en Java.

Es importante resaltar que el servicio está orientado sólo a brindar la comunicación necesaria para la transferencia de archivos pcap. Queda a cargo del auditor la instalación y utilización de herramientas, como Wireshark y Aircrack-ng, para poder llevar a cabo las tareas de análisis en el servidor.

El servicio se ejecuta sobre una PC configurada con la IP 192.168.1.100 y puerto 8080.

El código correspondiente al desarrollo del webservice está alojado en el respositorio Git correspondiente al proyecto.

4.2.3.3 Wireshark

Para verificar que la aplicación desarrollada genera de forma correcta los pcap del tráfico capturado, se utiliza la herramienta WireShark en su Versión 2.0.2 para examinar y analizar los datos de la captura.

4.2.3.4 Aircrack-ng

Aircrack-ng es una suite de software de seguridad inalámbrica. Consiste en un analizador de paquetes de redes, un crackeador de redes WEP y WPA/WPA2-PSK y otro conjunto de herramientas de auditoría inalámbrica. Se utiliza en las pruebas para crackear la clave WPA2 de la red analizada, a partir del pcap obtenido por la aplicación.

Comando utilizado para realizar el análisis de los paquetes:

```
https://gitlab.com/zube/pentestsecurityapp
```

4.2.3.5 Pentest Security App

Se instala la aplicación Pentest Security App en el Smartphone mencionado. Luego se configura la URL y el puerto del servicio web con los valores 192.168.1.100 y 8080 respectivamente.

4.3 Resultados

4.3.1 Comprobación de rooteo

A continuación se muestra el resultado de ejecutar la aplicación "Root Checker Basic" en el Motorola G de prueba, la cual permite corroborar que el dispositivo no está rooteado.

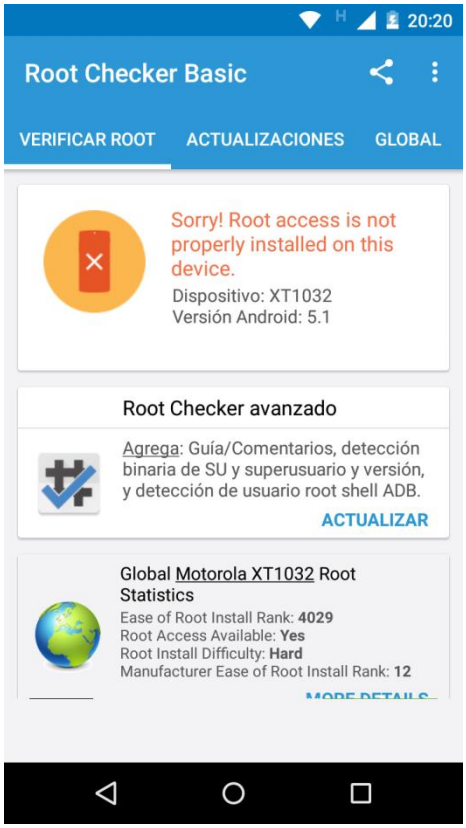


Fig. 4.18 – Aplicación que demuestra que el dispositivo no ha sido rooteado.

4.3.2 Consumo de batería

Se presentan cuatro (4) imágenes correspondientes al estado de la batería a medida que transcurría el tiempo de recopilación de datos. El estado de la misma se obtiene al marcar el código ***#*#4636#*#***, el cual despliega un menú con la información correspondiente.

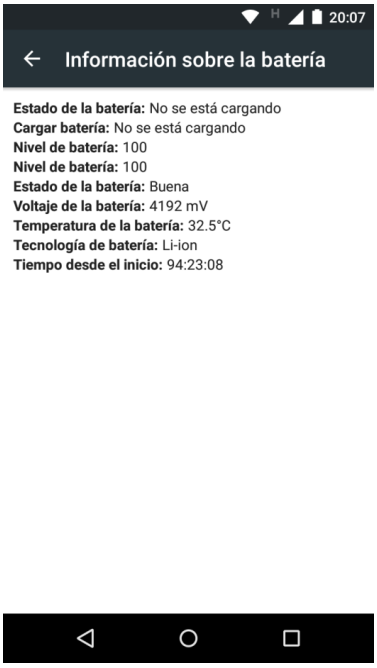


Fig. 4.19. Antes de comenzar el escaneo.

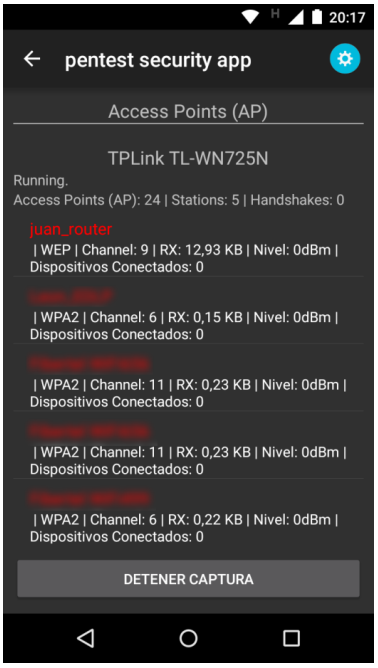


Fig. 4.20. Escaneo iniciado.

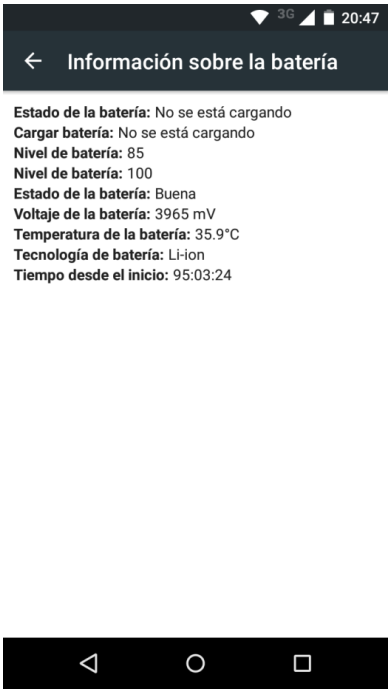


Fig. 4.21. Transcurridos 30 min del escaneo.

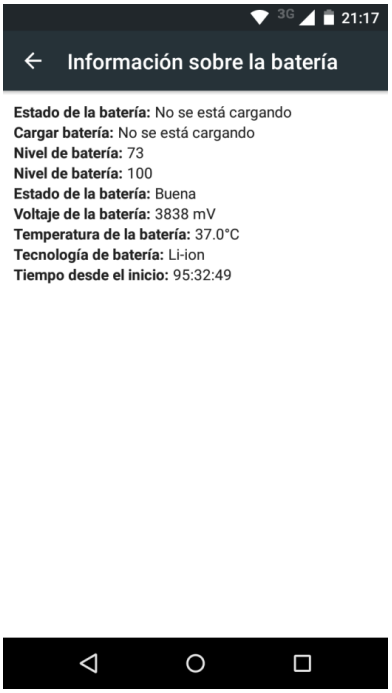


Fig. 4.22. Transcurridos 60 min del escaneo.

4.3.3 Análisis de Pcap con Wireshark

A continuación de muestra la Figura 4.23, donde se puede observar el resultado de abrir y analizar el pcap con Wireshark. En la línea que esta seleccionada se distingue el SSID "juan_router", dejando en evidencia el correcto funcionamiento de la aplicación Pentest Security App al momento de capturar el tráfico de red.

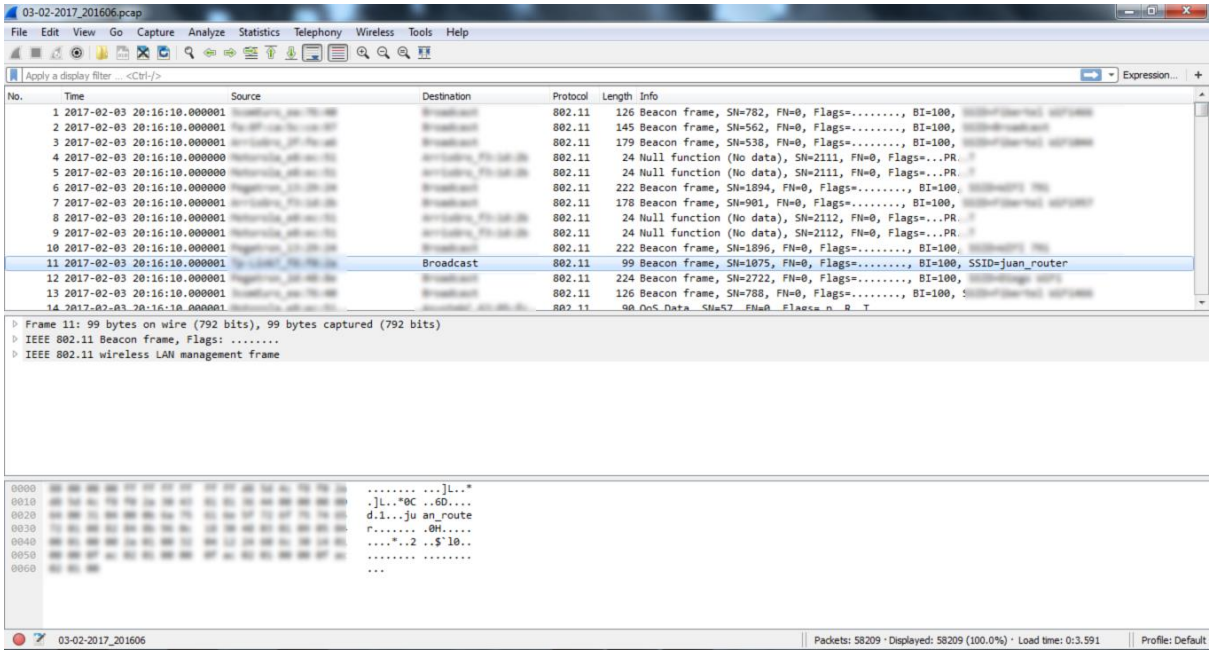
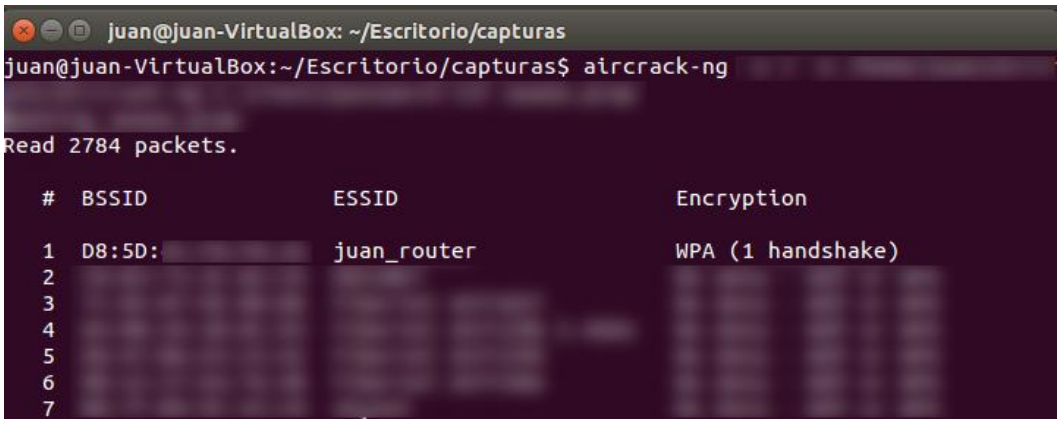


Fig. 4.23. Captura donde se puede observar el resultado del escaneo, con los paquetes de redes obtenidos. Wireshark.

4.3.4 Crackeo de clave WPA con Aircrack-ng

A continuación, en la Figura 4.24 se puede apreciar la detección de un handshake valido en el pcap analizado y en la Figura 4.25 se puede muestra el proceso exitoso de crakeo para obtener la clave que había sido configurada en el router, lo que deja en evidencia que la aplicación Pentest Security App, captura correctamente los frames de autenticación.



. 4.24. *Análisis del pcap obtenido de la aplicación en Aircrack-ng. Se puede observar el handshake capturado.*

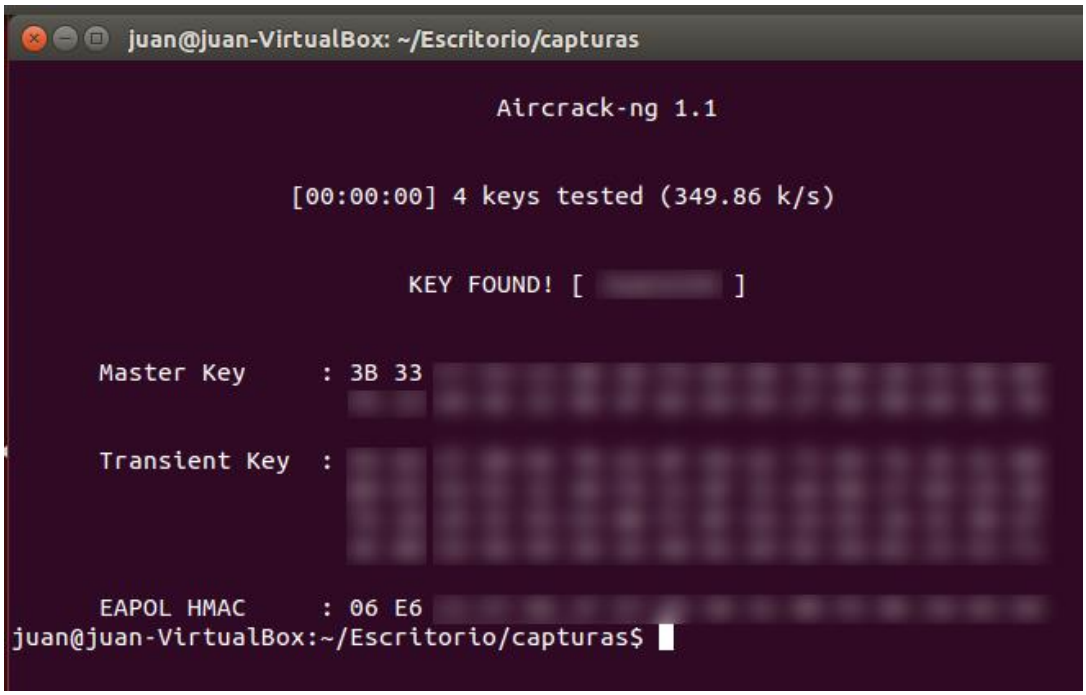


Fig. 4.25. Resultado final del proceso de escaneo.

Capítulo 5

5. Conclusión y Trabajo Futuro

5.1 Conclusión

El principal objetivo de la tesina fue desarrollar una herramienta que dé soporte a las tareas de relevamiento de redes inalámbricas en el contexto de una auditoría de seguridad, como una alternativa libre, práctica y cómoda que reúna gran parte de las funcionalidades que brindan en forma separada las opciones actuales.

Las auditorías de seguridad de redes inalámbricas resultan significativas para garantizar la integridad y el correcto funcionamiento de los sistemas informáticos. Llevarlas a cabo requiere descubrir los problemas de seguridad mediante pruebas de penetración o Pentest.

De las tareas que involucra un Pentest de redes inalámbricas, las actividades de relevamiento demandan, a la persona encargada de la auditoría, recorrer espacios físicos detectando e identificando los AP, siendo así el uso de una notebook poco práctico y generador de demoras asociadas a la dificultad de su traslado. Resulta ideal para esta etapa la utilización de smartphones que brindan portabilidad. No obstante, las soluciones actualmente disponibles presentan limitaciones y problemáticas ocasionando que no sean de gran aceptación entre los auditores.

Llevar a cabo el desarrollo de Pentest Security App requirió un trabajo de investigación teórico y práctico que tuvo como punto de inicio el diálogo con profesionales del área de seguridad en redes, quienes compartieron sus experiencias, forma de trabajo y opiniones de alto valor para que pudiésemos entender y dimensionar sus necesidades reales. La información obtenida, en conjunto con el estudio de las metodologías y buenas prácticas relacionadas con un Pentest de redes inalámbricas, permitió comprender y definir los criterios de evaluación para llevar a cabo el análisis de las aplicaciones existentes.

Las problemáticas identificadas en las aplicaciones analizadas se relacionan con el rooteo y cambio de firmware/ROM del dispositivo para obtener el modo monitor, el consumo de la placa wifi y la duración de la batería, y la suite de herramientas y su interfaz de consola. Habiendo identificado estas problemáticas se propusieron soluciones específicas para cada una, las cuales fueron tenidas en cuenta para el posterior desarrollo de la aplicación.

La aplicación móvil desarrollada en la presente tesina contempló la funcionalidad necesaria para llevar a cabo el conjunto de tareas más comunes de la etapa de relevamiento. La funcionalidad fue complementada con un servicio web para demostrar que las evidencias obtenidas con la aplicación móvil articulan correctamente con las tareas relacionadas a la siguiente etapa de un Pentest. A lo largo del desarrollo fueron muy importantes las herramientas y librerías open source, de las cuales se utilizó el código para implementar la solución final.

Para evaluar el funcionamiento de Pentest Security App se implementó un entorno de pruebas que permitió demostrar que la misma opera correctamente.

La aplicación desarrollada alcanza los objetivos planteados al inicio de la tesina, representando una solución superadora respecto de las aplicaciones preexistentes. Facilita y hace más seguro el proceso de instalación al prescindir del acceso root y la modificación del firmware del dispositivo. Hace un uso más responsable de la batería, combinando un chipset inalámbrico de bajo consumo eléctrico y aplicaciones de bajo procesamiento. Mejora la usabilidad al incorporar una interfaz nativa de Android y la portabilidad al utilizar placas inalámbricas de dimensiones reducidas. En conjunto todas estas mejoras contribuyen a que la solución brinde una mejor experiencia de usuario.

Respecto a la experiencia personal obtenida en el desarrollo de esta tesina, podemos destacar:

- Adquirimos mayor conocimiento en el desarrollo de aplicaciones móviles para Android.
- Comprendimos en detalle las problemáticas relacionadas con las auditorías de redes inalámbricas.
- Aprendimos sobre lo que implica abordar una problemática compleja, tanto desde un punto de vista teórico como práctico.

- Valoramos la importancia de las tecnologías open source, que nos permitieron alcanzar la aplicación implementada.

Con los resultados obtenidos, se espera haber hecho un aporte a la comunidad académica del área de la informática y a los profesionales del área de seguridad en redes.

5.2 Trabajo Futuro

A partir del desarrollo realizado se sugieren los siguientes trabajos futuros:

- Incorporar la posición geográfica de los AP, reflejando los resultados en un plano edilicio del lugar auditado.
- Incorporar nuevos drivers de placas inalámbricas externas para abarcar una mayor cantidad de dispositivos soportados por la aplicación (al momento de realizar el trabajo solo se pueden utilizar aquellas que tengan chipset RTL8188CU, RTL8192CU).
- Desarrollar web services que brinden funcionalidad para llevar a cabo el resto de las etapas de un pentest de redes inalámbricas.
- Incorporar otros formatos de presentación de la información obtenida.

6. Bibliografía

1. WIFI. [En línea] http://www.wi-fi.org/.

2. **Ramachandran, Vivek.** *BackTrack 5 Wireless Penetration Testing Beginner's Guide.* 2011.

3. Open Web Application Security Project . [En línea] https://www.owasp.org.

4. Certified Ethical Hacking. [En línea] https://www.eccouncil.org/programs/certified-ethical-hacker-ceh/.

5. Open Source Security Testing Methodology Manual. [En línea] http://www.isecom.org/research/.

6. Information Systems Security Assessment Framework. [En línea] https://www.owasp.org/index.php/Penetration_testing_methodologies.

7. Penetration Testing Execution Standard. [En línea] https://www.owasp.org/index.php/Penetration_testing_methodologies#Penetration_Testing_Execution_Standard_.28PTES.29.

8. Access Point. [En línea] http://www.wi-fi.org.

9. Universidad Nacional de La Plata. [En línea] www.unlp.edu.ar/.

10. **Gonzalez, Pablo, Sanchez, German y Soriano, Jose Miguel.** *Pentesting con Kali Linux.* 2013.

11. **Federico G. Pacheco, Hector Jara.** *Ethical Hacking 2.0.* 2012.

12. Rogue Access Points. [En línea] http://www.wi-fiplanet.com/tutorials/article.php/1564431.

13. **Catoira, Fernando.** Hackers en la red. [En línea] Agosto de 2012. https://hackersenlared.wordpress.com/category/capacitacion/que-es-un-pentest/.

14. **Mizrachi, Ing. Aarón.** *Guía de Pentesting Básico.* 2014.

15. **Ec-Council.** *Penetration Testing: Communication Media Testing.* 2010.

16. WPA. [En línea] http://www.ieee802.org/11/.

17. WPA2. [En línea] http://www.ieee802.org/11/.

18. Iphone. [En línea] http://www.gsmarena.com/apple_iphone-1827.php.

19. Modo Monitor. [En línea]
http://searchsecurity.techtarget.com/definition/promiscuous-mode.

20. Nokia N900. [En línea] http://www.smart-gsm.com/moviles/nokia-n900.

21. kismet. [En línea] https://www.kismetwireless.net/documentation.shtml.

22. Nokia. [En línea] http://www.nokia.com/es_int/acerca-de-nokia.

23. **Arena, GSM.** [En línea] http://www.gsmarena.com/nokia_n900-2917.php.

24. **Maemo.org.** [En línea] http://maemo.org/intro/.

25. Kismet For Nokia N900. [En línea]
https://david.gnedt.at/blog/2010/05/11/kismet-fully-functional-monitor-mode-for-the-n900/.

26. tuexpetoMovil.com. [En línea] 2010.
https://www.tuexpertomovil.com/2010/11/05/samsung-galaxy-s-iphone-4-desbancado-en-ventas-por-el-samsung-galaxy-s-en-japon/.

27. **Virki, Tarmo.** Reuters. [En línea] 2010.
http://www.reuters.com/article/us-nokia-n-idUSTRE64R1DI20100528.

28. Maemo. [En línea] http://wiki.maemo.org.

29. David Gnedt Blog. [En línea] https://david.gnedt.at/blog/.

30. Nokia Video 1. [En línea]
https://www.youtube.com/watch?v=Mzml8eAxhvE.

31. Nokia N900 Video 2. [En línea]
https://www.youtube.com/watch?v=KvalGUYWjuQ.

32. Telegraph. [En línea]
http://www.telegraph.co.uk/technology/2016/01/26/the-20-bestselling-mobile-phones-of-all-time/.

33. BcMon Blogspot. [En línea] http://bcmon.blogspot.com.ar/2012/09/working-monitor-mode-on-nexus-one.html.

34. CyanogenMod. [En línea] https://github.com/CyanogenMod.

35. **Feinstein, Ruby.** [En línea] http://bcmon.blogspot.com.ar/.

36. Tutorial Bcmon. [En línea] http://hackagon.com/hack-wifi-using-android-phones/.

37. Video Tutorial Bcmon. [En línea] https://www.youtube.com/watch?v=_lwIG9pP2Dw.

38. Efuse. [En línea] http://www.androidpolice.com/2015/11/06/why-does-my-android-phone-have-efuses-and-why-should-i-care-about-them/.

39. Samsung. [En línea] http://www.samsung.com/ar/support/model/GT-I9100LKPUFN.

40. Kingo Root. [En línea] https://www.kingoapp.com/root-samsung.htm.

41. **Rodrigo, German.** AZone. [En línea] http://androidzone.org/2012/02/z4root-para-android-ser-root-nunca-fue-tan-facil-apk/.

42. **Antonio, Jose.** rootear. [En línea] rootear.com/android/rootear-samsung-galaxy-s-ii.

43. Android Studio. [En línea] https://developer.android.com/studio/command-line/adb.html?hl=es-419.

44. clockworkmod. [En línea] http://www.clockworkmod.com/.

45. Team Win Recovery Project. [En línea] https://twrp.me/.

46. Aircrack-ng. [En línea] https://www.aircrack-ng.org/doku.php?id=es:airodump-ng.

47. **WirelessHac.** WirelessHac. [En línea] http://www.wirelesshack.org/how-to-use-the-wash-command-to-find-wps-enabled-routers-with-backtrack-5-or-kali-linux.html.

48. wireshark. [En línea] https://www.wireshark.org/.

49. eyePa. [En línea] http://www.metageek.com/products/eye-pa/.

50. **Atanasov, Vencislav.** [En línea] http://kismetwireless.net/gitweb/?p=android-pcap.git;a=commitdiff;h=8c9160b0c28c599ef0a8abb6e2b0c459ec6fe70f.

51. Android Pcap. [En línea] https://www.kismetwireless.net/android-pcap/.

52. AndroidPcacpTutorial 1. [En línea] https://www.youtube.com/watch?v=zNfbeB0Urvc.

53. Android Pcap tutorial 2. [En línea] https://www.youtube.com/watch?v=fyQE9v5lBAg.

54. [En línea] https://www.youtube.com/watch?v=GiFN3wH3QFs.

55. Android Open Pwn Project. [En línea] https://www.pwnieexpress.com/blog/android-open-pwn-project.

56. Android Open Source Project . [En línea] https://source.android.com/.

57. kali Linux. [En línea] https://www.kali.org/.

58. Pwn Phone. [En línea] https://store.pwnieexpress.com/product/pwn-phone2014b/.

59. PwnPhone 2009 Nokia N900. [En línea] https://www.pwnieexpress.com/products/pentesting-community-editions?___hssc=214639368.32.1480537597315&___hstc=214639368.a531c28f5d21484a1d6f388af23f6178.1480363768745.1480375395708.1480537597315.4&___hsfp=1467042889&hsCtaTracking=98cca6b5-252a-4daa-9d7b-ae9.

60. Google Nexus 7. [En línea] https://www.asus.com/latin/Tablets/Nexus_7/.

61. PwnIEExpress. [En línea] https://www.pwnieexpress.com/products/pentesting-community-editions?___hssc=214639368.32.1480537597315&___hstc=214639368.a531c28f5d21484a1d6f388af23f6178.1480363768745.1480375395708.1480537597315.4&___hsfp=1467042889&hsCtaTracking=98cca6b5-252a-4daa-9d7b-ae9.

62. PwnPhone Dispositivos Soportados. [En línea] https://wiki.pwnieexpress.com/index.php/Supported_Devices.

63. Google Nexus 4. [En línea] http://www.smart-gsm.com/moviles/lg-nexus-4.

64. Google Nexus 5. [En línea] http://www.smart-gsm.com/moviles/google-nexus-5.

65. kali Linux Nethunter/. [En línea] https://www.kali.org/kali-linux-nethunter/.

66. streaming. [En línea] http://www.ite.educacion.es/formacion/materiales/107/cd/video/video0103.html.

67. **Moreno, Máximo.** Rootear. [En línea] https://rootear.com/android/riesgos-hacer-root.

68. **Celeiro, Òscar.** Andro4all. [En línea] https://andro4all.com/2013/02/10-razones-no-obtener-root.

69. WikiDevi - Chipset RTL8188CUS. [En línea] https://wikidevi.com/w/index.php?title=Special%3AAsk&q=%5B%5BChip1+model%3A%3ARTL8188CUS%5D%5D&po=%3FInterface%0D%0A%3FForm+factor%3DFF%0D%0A%3FInterface+connector+type%3DUSB+conn.%0D%0A%3FFCC+ID%0D%0A%3FManuf%0D%0A%3FManuf+product+model%3DManuf.+mdl%0D%0.

70. WikiDevi - Chipset RTL8192CU. [En línea] https://wikidevi.com/w/index.php?title=Special:Ask&offset=0&limit=500&q=%5B%5BChip1+model%3A%3ARTL8192CU%5D%5D&p=format%3Dbroadtable%2Flink%3Dall%2Fheaders%3Dshow%2Fsearchlabel%3D%E2%80%A6-20further-20results%2Fclass%3Dsortable-20wikitable-20smwtable&po=%.

71. Github. [En línea] https://github.com/gat3way/AirPirate/blob/master/src/com/gat3way/airpirate/Rtl8192Card.java.

72. Estadisticas Android. [En línea] https://developer.android.com/about/dashboards/index.html?hl=es.

73. Todo Android. [En línea] http://www.todoandroid.es/index.php/faq-de-android/65-versiones/1698-android-studio-o-eclipse-opinion-de-un-desarrollador-de-aplicaciones.html.

74. **Perez, Feddy.** [En línea] http://www.feddyperez.com/2015/11/por-que-usar-android-studio-para.html.

75. i2factory. [En línea] http://www.i2factory.com/es/integracion/qu%C3%A9-es-un-servicio-restful.

76. **Silva, Diego.** [En línea] http://www.apuntesdejava.com/2010/11/restful-la-forma-mas-ligera-de-hacer.html.

77. **Seta, Leonardo De.** Dos Ideas. [En línea] https://dosideas.com/noticias/java/314-introduccion-a-los-servicios-web-restful.html.

78. **Baez, Luis y Ehlen, Luis.** Programación de Sistemas. [En línea] http://profesores.elo.utfsm.cl/~agv/elo330/2s03/projects/Tomcat/.

79. [En línea] https://www.youtube.com/watch?v=zNfbeB0Urvc.

80. [En línea] https://www.youtube.com/watch?v=fyQE9v5lBAg.

81. Pwn Pad. [En línea] https://store.pwnieexpress.com/product/pwn-pad-4/.